



**ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**ESCUELA DE INGENIERÍA ELECTRÓNICA EN TELECOMUNICACIONES**  
**Y REDES**

**“DISEÑO E IMPLEMENTACIÓN DE UNA PANTALLA DIGITAL**  
**INTERACTIVA EMULADA UTILIZANDO TECNOLOGÍA**  
**INFRAROJA COMO UN RECURSO DIDÁCTICO PARA IPREX”**

Trabajo de titulación presentado para optar al grado académico de:  
**INGENIERO EN ELECTRÓNICA, TELECOMUNICACIONES Y**  
**REDES**

**AUTORES: JORGE LUIS ZÚÑIGA MAYORGA**  
**EDDY FABIAN GUSQUI GULLSQUI**

**TUTOR: ING. OSWALDO GEOVANNY MARTINEZ GUASHIMA**

Riobamba – Ecuador

2017

©**2017**, Jorge Luis Zúñiga Mayorga; Eddy Fabián Gusqui Gullsqui.

Se autoriza la reproducción total o parcial, con fines académicos, por cualquier medio o procedimiento, incluyendo la cita bibliográfica del documento, siempre y cuando se reconozca el Derecho de Autor.

**ESCUELA SUPERIOR POLITÉCNICA DE CHIMBORAZO**  
**FACULTAD DE INFORMÁTICA Y ELECTRÓNICA**  
**ESCUELA DE INGENIERÍA ELECTRÓNICA EN TELECOMUNICACIONES Y**  
**REDES**

El Tribunal del Trabajo de Titulación certifica que: El trabajo de investigación **“DISEÑO E IMPLEMENTACIÓN DE UNA PANTALLA DIGITAL INTERACTIVA EMULADA UTILIZANDO TECNOLOGÍA INFRAROJA COMO UN RECURSO DIDÁCTICO PARA IPREX”**, de responsabilidad de los señores: Eddy Fabián Gusqui Gullsqui y Jorge Luis Zúñiga Mayorga, ha sido minuciosamente revisado por los Miembros del Tribunal de Trabajo de Titulación, quedando autorizada su presentación.

<b>NOMBRE</b>	<b>FIRMA</b>	<b>FECHA</b>
Ing. Washington Luna, Msc. <b>DECANO DE LA FACULTAD DE INFORMÁTICA Y ELECTRÓNICA</b>	_____	_____
Ing. Franklin Moreno, Msc. <b>DIRECTOR DE ESCUELA DE ELECTRÓNICA, TELECOMUNICACIONES Y REDES</b>	_____	_____
Ing. Oswaldo Martínez, Msc. <b>DIRECTOR DE TRABAJO DE TITULACIÓN</b>	_____	_____
Dr. Mario Paguay <b>MIEMBRO DEL TRIBUNAL</b>	_____	_____

Nosotros, JORGE LUIS ZÚÑIGA MAYORGA y EDDY FABIAN GUSQUI GULLSQUI, somos responsables de las ideas, doctrinas y resultados expuestos en este Trabajo de Titulación, y el patrimonio intelectual del mismo le pertenecen a la Escuela Superior Politécnica de Chimborazo

-----  
Jorge Luis Zúñiga Mayorga

CI: 180497280-8

-----  
Eddy Fabián Gusqui Gullsqui

CI: 070628255-5

## **DEDICATORIA**

Agradezco a Dios por permitirme llegar hasta un nuevo objetivo, cuidándome y dándome fortaleza para continuar, y a mi familia, quienes en el transcurso de mi vida han velado por mi bienestar y educación brindándome su apoyo incondicional.

Eddy Fabián

A Dios por regalarme la vida, guiarme y bendecirme siempre.

A toda mi familia por el apoyo incondicional, gracias a sus buenos consejos que me ayudaron a formar como persona.

Jorge Luis

## **AGRADECIMIENTO**

Agradecemos a Dios por guiarnos día a día durante toda nuestra carrera académica ya que gracias a él logramos hacer realidad esta meta tan anhelada. También agradecemos a nuestros padres por ese apoyo incondicional que supieron brindarnos, por sus consejos y sugerencias las cuales motivaron para salir adelante

A nuestra familia que son el pilar fundamental, gracias por sus palabras de aliento y por ser siempre solidarios, a nuestros amigos que formaron parte importante de nuestras vidas, gracias por su amistad por todos los momentos buenos y malos que pasamos porque de ello aprendimos muchas cosas.

A los docentes de la Facultad de Informática y electrónica, especialmente a nuestro director del trabajo de titulación Ing. Oswaldo Martínez por su apoyo incondicional, paciencia y motivación, ya que su enseñanza académica y moral han sido de gran ayuda para culminar este trabajo.

Jorge, Eddy

## TABLA DE CONTENIDO

<b>PORTADA</b>	i
<b>DERECHOS DE AUTOR</b>	ii
<b>CERTIFICACIÓN</b>	iii
<b>DECLARACIÓN DE RESPONSABILIDAD</b>	iv
<b>DEDICATORIA</b>	v
<b>AGRADECIMIENTO</b>	vi
<b>TABLA DE CONTENIDO</b>	vii
<b>INDICE DE FIGURAS</b>	xii
<b>INDICE DE TABLAS</b>	xv
<b>ÍNDICE DE GRAFICOS</b>	xvi
<b>ÍNDICE DE ANEXOS</b>	xvii
<b>INDICE DE ABREBIATURAS</b>	xviii
<b>RESUMEN</b>	xx
<b>SUMMARY</b>	xxi
<b>INTRODUCCIÓN</b>	1
<b>ANTECEDENTES</b>	1
<b>FORMULACIÓN DEL PROBLEMA</b>	3
<b>SISTEMATIZACIÓN DEL PROBLEMA</b>	3
<b>JUSTIFICACIÓN TEÓRICA</b>	4
<b>JUSTIFICACIÓN APLICATIVA</b>	5
<b>OBJETIVOS</b>	5
Objetivos Generales	5
Objetivos Específicos	6
<b>CAPITULO I</b>	
<b>1 MARCO REFERENCIAL</b>	7
<b>1.1 Pantalla Digital Interactiva</b>	7
<i>1.1.1 Pizarra Digital Interactiva Portátil (PDIP)</i>	8
<i>1.1.2 Pizarras De Pantalla Frontal</i>	9
<i>1.1.3 Pizarra de Barras de Coordenadas o Ángulos</i>	9
<i>1.1.4 Pantallas Planas Táctiles</i>	10
<i>1.1.5 Tecnologías de las PDI</i>	10
<i>1.1.5.1 Electromagnética</i>	10
<i>1.1.5.2 Resistiva</i>	10

1.1.5.3	Óptica .....	11
1.1.5.4	Infrarroja .....	11
1.1.6	<b>Elementos que integran la PDI</b> .....	11
1.2	<b>Wiimote</b> .....	12
1.2.1	<b>Funcionamiento del mando Wii</b> .....	13
1.2.2	<b>Direcciones de memoria que recibe el mando Wii</b> .....	14
1.2.2.1	Puertos y estados .....	14
1.2.3	<b>Componentes internos del mando Wii</b> .....	15
1.2.3.1	Acelerómetro .....	15
1.2.3.2	Cámara Infrarroja .....	16
1.2.3.2.1	Características .....	17
1.2.3.2.2	Activación de la cámara .....	17
1.2.3.2.3	Configuración de sensibilidad .....	18
1.2.3.2.4	Formato de los datos .....	19
1.2.3.2.4.1	Modo Básico .....	19
1.2.3.2.4.2	Modo Extendido .....	20
1.2.3.2.4.3	Modo Completo .....	20
1.2.3.3	Chip emisor/receptor Bluetooth .....	21
1.2.4	<b>Componentes externos del mando Wii</b> .....	21
1.2.4.1	Nunchuk .....	21
1.2.4.1.1	Registros e inicialización .....	22
1.2.4.1.2	Registros e inicialización de Identificación .....	22
1.2.4.1.3	Información De Estado .....	22
1.2.4.1.4	Movimientos del Nunchuk .....	23
1.2.4.1.5	Partes del Nunchuk .....	24
1.3	<b>Lenguajes de Programación</b> .....	25
1.3.1	<b>LabView</b> .....	25
1.3.1.1	Características .....	25
1.3.1.2	Aspectos generales .....	26
1.3.1.2.1	Panel frontal .....	26
1.3.1.2.2	Diagrama de Bloque .....	26
1.3.1.2.3	Compilación del programa .....	27
1.3.1.2.4	Paletas .....	27
1.3.1.2.4.1	Paletas de control .....	27
1.3.1.2.4.2	Paleta de funciones .....	27
1.3.1.2.4.3	Paleta de Herramientas .....	28
1.3.1.2.5	Comunicación serial .....	28



1.3.1.2.5.1	VISA.....	28
1.3.1.2.6	Visa Configure Serial Port .....	29
1.3.1.2.6.1	VISA lectura.....	30
1.3.1.2.6.2	VISA cierre .....	30
1.3.1.2.6.3	Nodo de propiedad .....	30
1.3.1.3	Estructuras de LabVIEW.....	31
1.3.1.4	Call Library Function Node .....	32
1.3.1.5	System exec.vi .....	33
<b>1.3.2</b>	<b>Visual Studio C# .....</b>	<b>33</b>
1.3.2.1	Características .....	34
1.3.2.2	Funcionalidades. ....	34
1.3.2.3	Elementos Importantes .....	35
1.3.2.4	Librería WiimoteLib .....	36
1.3.2.4.1	DataTypes.cs .....	36
1.3.2.4.2	Events.cs .....	37
1.3.2.4.3	HIDImports.cs .....	37
1.3.2.4.4	Wiimote.cs .....	38
1.3.2.4.5	WiimoteCollection.cs .....	38

## **CAPITULO II**

<b>2</b>	<b>MARCO METODOLÓGICO .....</b>	<b>38</b>
<b>2.1</b>	<b>Introducción.....</b>	<b>39</b>
2.1.1	Métodos de investigación.....	39
<b>2.2</b>	<b>Selección de Hardware.....</b>	<b>40</b>
2.2.1	Análisis y elección del hardware .....	40
2.2.1.1	Selección del dispositivo infrarrojo.....	40
2.2.1.1.1	Ponderación .....	40
2.2.1.2	Parámetros concernientes .....	41
2.2.1.2.1	Precio .....	41
2.2.1.2.2	Características .....	41
2.2.1.2.3	Actualización y mejoras del dispositivo .....	42
2.2.1.3	Resultados en la elección del Hardware .....	42
<b>2.2.2</b>	<b>Etapa Hardware.....</b>	<b>43</b>
2.2.2.1	Wii Motion Plus .....	44
2.2.2.2	Nunchuk.....	44
2.2.2.3	Arduino Nano .....	45
2.2.2.4	Módulos Xbee .....	46
2.2.2.5	Adaptador Bluetooth SCR .....	46

2.2.2.6	<i>Puntero Infrarrojo</i> .....	47
<b>2.3</b>	<b>Selección de Software</b> .....	47
<b>2.3.1</b>	<b><i>Elección de Software y plataformas a usar</i></b> .....	47
<b>2.3.2</b>	<b><i>Etapas de software</i></b> .....	48
2.3.2.1	<i>Visual Studio C# 2015</i> .....	48
2.3.2.2	<i>LabView 2015</i> .....	48
2.3.2.3	<i>Arduino Nightly</i> .....	49
2.3.2.4	<i>Programa XCTU</i> .....	49
<b>2.4</b>	<b>Diseño del Algoritmo</b> .....	49
<b>2.4.1</b>	<b><i>Introducción</i></b> .....	49
<b>2.4.2</b>	<b><i>Desarrollo de la Pizarra Digital Interactiva</i></b> .....	50
2.4.2.1	<i>Diseño del puntero Infrarrojo</i> .....	50
2.4.2.2	<i>Conexión del Mando Wii con el Ordenador</i> .....	51
2.4.2.2.1	<i>Computador Portátil</i> .....	51
2.4.2.2.2	<i>Computadora de Escritorio (PC)</i> .....	53
2.4.2.3	<i>Desarrollo de la Aplicación de la PDI</i> .....	54
<b>2.4.3</b>	<b><i>Desarrollo del control inalámbrico</i></b> .....	57
2.4.3.1	<i>Conexión del Nunchuk con el Ordenador</i> .....	57
2.4.3.1.1	<i>Instalación del Software Arduino</i> .....	57
2.4.3.1.2	<i>Programación en el Nano Arduino</i> .....	58
2.4.3.1.2.1	<i>Incorporación de la librería WiiChuck.h</i> .....	58
2.4.3.1.2.2	<i>Programación</i> .....	58
2.4.3.1.3	<i>Vinculación de los módulos Xbee</i> .....	59
2.4.3.1.3.1	<i>Conexión física del emisor/receptor</i> .....	59
2.4.3.1.3.2	<i>Configuración del sistema</i> .....	60
2.4.3.2	<i>Desarrollo de la Aplicación del Control Inalámbrico de la PDI</i> .....	63
<b>2.4.4</b>	<b><i>Desarrollo de la Aplicación Final</i></b> .....	71
2.4.4.1	<i>Integración entre las Aplicaciones de la PDI y el Control Inalámbrico</i> .....	72
2.4.4.2	<i>Generación de la aplicación ejecutable</i> .....	73
2.4.4.3	<i>Interfaz gráfica</i> .....	76
<b>CAPITULO III</b>		
<b>3</b>	<b>MARCO DE RESULTADOS, ANÁLISIS Y VALIDACIÓN</b> .....	77
<b>3.1</b>	<b>Introducción</b> .....	77
<b>3.2</b>	<b>Instalación del software de la PDI y Control Inalámbrico</b> .....	77
<b>3.3</b>	<b>Funcionamiento del sistema de la pantalla digital interactiva</b> .....	78
<b>3.3.1</b>	<b><i>Funcionamiento de la Pantalla Interactiva</i></b> .....	79
<b>3.3.2</b>	<b><i>Funcionamiento del Control Inalámbrico</i></b> .....	79

<b>3.4</b>	<b>Implementación .....</b>	<b>80</b>
<b>3.4.1</b>	<b><i>Vinculación del Mando con el ordenador .....</i></b>	<b>81</b>
<b>3.4.2</b>	<b><i>Manipulación del software .....</i></b>	<b>82</b>
<b>3.5</b>	<b>Resultados .....</b>	<b>84</b>
<b>3.5.1</b>	<b><i>Determinación de la ubicación del Mando Wii .....</i></b>	<b>85</b>
<b>3.5.2</b>	<b><i>Pruebas Realizadas .....</i></b>	<b>86</b>
<b>3.6</b>	<b>Análisis de Resultados .....</b>	<b>89</b>
<b>3.7</b>	<b>Validación de Resultados .....</b>	<b>92</b>
<b>3.7.1</b>	<b><i>Por sus características .....</i></b>	<b>92</b>
<b>3.7.2</b>	<b><i>Por su precio .....</i></b>	<b>93</b>
<b>3.7.3</b>	<b><i>Por su eficiencia y precisión.....</i></b>	<b>96</b>
<b>3.7.4</b>	<b><i>Por su nivel de aceptación .....</i></b>	<b>99</b>
	<b>CONCLUSIONES .....</b>	<b>102</b>
	<b>RECOMENDACIONES .....</b>	<b>103</b>
	<b>BIBLIOGRAFÍA</b>	
	<b>ANEXOS</b>	

## INDICE DE FIGURAS

<b>Figura 1-1:</b> Diseño de la PDI .....	7
<b>Figura 2-1:</b> PDI con dispositivos de control. ....	8
<b>Figura 3-1:</b> Pizarras MIMO .....	9
<b>Figura 4-1:</b> Pantalla digital frontal o retroalimentada .....	9
<b>Figura 5-1:</b> Pantalla de barra de ángulos .....	10
<b>Figura 6-1:</b> Pantalla LCD, Plasma interactivas .....	10
<b>Figura 7-1:</b> Mando Wii Remote Plus .....	12
<b>Figura 8-1:</b> Acelerómetro .....	16
<b>Figura 9-1:</b> Cámara del mando Wii .....	17
<b>Figura 10-1:</b> Ubicación de la cámara del mando Wii .....	17
<b>Figura 11-1:</b> Formato de datos en modo básico .....	20
<b>Figura 12-1:</b> Formato de datos en modo extendido. ....	20
<b>Figura 13-1:</b> Formato de datos en modo completo .....	21
<b>Figura 14-1:</b> Chip Bluetooth del mando Wii .....	21
<b>Figura 15-1:</b> Movimiento hacia la derecha o izquierda. ....	23
<b>Figura 16-1:</b> Movimiento hacia adelante o atrás .....	24
<b>Figura 17-1:</b> Movimiento hacia arriba o abajo .....	24
<b>Figura 18-1</b> Partes del Nunchuk .....	25
<b>Figura 19-1:</b> Panel Frontal. ....	26
<b>Figura 20-1:</b> Diagrama de bloque .....	26
<b>Figura 21-1:</b> Paleta de control .....	27
<b>Figura 22-1:</b> Paleta de funciones .....	28
<b>Figura 23-1:</b> Paleta de herramientas .....	28
<b>Figura 24-1:</b> Puerto serial VISA .....	29
<b>Figura 25-1:</b> VISA configure Serial Port .....	29
<b>Figura 26-1</b> VISA de lectura .....	30
<b>Figura 27-1:</b> VISA de cierre .....	30
<b>Figura 28-1:</b> Detalles del nodo de propiedad .....	31
<b>Figura 29-1:</b> Ciclo While .....	31
<b>Figura 30-1:</b> Ciclo For .....	32
<b>Figura 31-1:</b> Call Library Function Node .....	32
<b>Figura 32-1:</b> System Exec.vi .....	33
<b>Figura 33-1:</b> Plataforma del Visual Studio C# .....	35
<b>Figura 1-2</b> Resultado para elección de dispositivos .....	43

<b>Figura 2-2:</b> Bus I2C del Nunchuk .....	45
<b>Figura 3-2:</b> Partes internas de Nunchuk.....	45
<b>Figura 4-2:</b> Arduino Nano y diagrama de pines.....	45
<b>Figura 5-2:</b> Módulo Xbee y diagrama de pines.....	46
<b>Figura 6-2:</b> Adaptador de Bluetooth .....	47
<b>Figura 7-2:</b> Puntero Infrarrojo.....	50
<b>Figura 8-2:</b> Búsqueda de dispositivos Bluetooth .....	51
<b>Figura 9-2:</b> Asistente de configuración Bluetooth. ....	51
<b>Figura 10-2:</b> Botón SYN de Mando Wii.....	52
<b>Figura 11-2:</b> Selección de dispositivo Bluetooth .....	52
<b>Figura 12-2:</b> Dispositivo Bluetooth sincronizado .....	53
<b>Figura 13-2:</b> Adaptador Bluetooth CSR.....	53
<b>Figura 14-2:</b> Instalación del controlador de Bluetooth .....	53
<b>Figura 15-2:</b> Búsqueda del Mando Wii.....	54
<b>Figura 16-2:</b> Emparejamiento del dispositivo.....	54
<b>Figura 17-2:</b> Instalación del software Arduino .....	58
<b>Figura 18-2:</b> Instalación de la librería WiiChuck.h.....	58
<b>Figura 19-2:</b> Conexión del Nunchuk, Arduino y Xbee .....	60
<b>Figura 20-2:</b> Vinculación con X-CTU .....	61
<b>Figura 21-2:</b> Verificación Test/Query.....	61
<b>Figura 22-2:</b> Función Read del XB1 .....	62
<b>Figura 23-2:</b> Funcion Read del XB2.....	62
<b>Figura 24-2:</b> Modulo Emisor.....	63
<b>Figura 25-2:</b> Modulo Receptor.....	63
<b>Figura 26-2:</b> Interfaz de la herramienta VISA. ....	65
<b>Figura 27-2:</b> Parámetros de VISA Configure Serial Port.....	65
<b>Figura 28-2:</b> Unión de Visa Configure Serial Port y Property Node .....	66
<b>Figura 29-2:</b> Sesión VISA .....	66
<b>Figura 30-2:</b> Integración del Decimal String to Number .....	67
<b>Figura 31-2:</b> Case Structure para el eje X/Y .....	67
<b>Figura 32-2:</b> Incorporación de Call Library Function Node. ....	68
<b>Figura 33-2:</b> Case Structure para los botones C y Z .....	68
<b>Figura 34-2:</b> Configuración para el evento C .....	69
<b>Figura 35-2:</b> Configuración para el evento Z.....	69
<b>Figura 36-2:</b> Integración de la instrucción While Loop.....	70
<b>Figura 37-2:</b> Interfaz Gráfica para verificación de errores .....	70
<b>Figura 38-2:</b> Programación del control inalámbrico.....	71

<b>Figura 39-2:</b> Vinculación con la aplicación de Control Inalámbrico .....	72
<b>Figura 40-2:</b> Vinculación con la aplicación de la PDI .....	72
<b>Figura 41-2:</b> Programación de aplicación final .....	73
<b>Figura 42-2</b> Modificación del programa principal .....	73
<b>Figura 43-2:</b> Creación de un nuevo proyecto .....	74
<b>Figura 44-2:</b> Configuración para una aplicación .EXE .....	74
<b>Figura 45-2:</b> Configuración de parámetros de información .....	75
<b>Figura 46-2:</b> Vinculación con programa principal .....	75
<b>Figura 47-2:</b> Proceso para guardar archivo ejecutable .....	75
<b>Figura 48-2:</b> Archivo ejecutable .....	76
<b>Figura 49-2:</b> Interfaz Gráfica de la Aplicación .....	76
<b>Figura 1-3:</b> Ubicación de archivos .....	78
<b>Figura 2-3:</b> Etapas de aplicación .....	78
<b>Figura 3-3:</b> Opción de calibración .....	79
<b>Figura 4-3:</b> Diagrama de bloques del control inalámbrico .....	80
<b>Figura 5-3:</b> Esquema del Prototipo .....	81
<b>Figura 6-3:</b> Vinculación Bluetooth .....	81
<b>Figura 7-3:</b> Portada de la aplicación .....	82
<b>Figura 8-3:</b> Ejecución de la aplicación de la PDI .....	82
<b>Figura 9-3:</b> Calibración de la PDI .....	83
<b>Figura 10-3:</b> Manipulación de la PDI .....	83
<b>Figura 11-3:</b> Ejecución de la aplicación del control inalámbrico .....	84
<b>Figura 12-3</b> Manipulación de control inalámbrico .....	84
<b>Figura 13-3:</b> Lugares de prueba para el sistema .....	85
<b>Figura 14-3:</b> Calculo de distancia para el Mando Wii .....	85
<b>Figura 15-3:</b> Ejemplos resueltos de ubicación del Mando Wii .....	86
<b>Figura 16-3:</b> Fijación del Mando Wii en la base .....	86
<b>Figura 17-3:</b> Calibración de la pantalla .....	87
<b>Figura 18-3:</b> Función de arrastre .....	87
<b>Figura 19-3:</b> Comprobación del puntero .....	88
<b>Figura 20-3:</b> Función clic derecho del control .....	88
<b>Figura 21-3:</b> Función clic izquierdo del control .....	89
<b>Figura 22-3:</b> Ejemplo de pantallas mal proyectadas .....	89
<b>Figura 23-3:</b> Ángulos y distancia de funcionamiento del Mando Wii .....	90
<b>Figura 24-3:</b> Enfoque a 45° .....	90
<b>Figura 25-3:</b> Enfoque a 75° .....	91
<b>Figura 26-3:</b> Enfoque a 90° .....	92

## INDICE DE TABLAS

<b>Tabla 1-1:</b> Direcciones de memoria del Mando Wii .....	14
<b>Tabla 2-1:</b> Puertos y estados del Mando Wii .....	14
<b>Tabla 3-1:</b> Registros de Sensibilidad .....	18
<b>Tabla 4-1:</b> Inicialización e Identificación .....	22
<b>Tabla 5-1:</b> Información de estado .....	23
<b>Tabla 1-2:</b> Ponderación para elección de hardware .....	41
<b>Tabla 2-2:</b> Precios de los dispositivos .....	41
<b>Tabla 3-2:</b> Características de los dispositivos .....	42
<b>Tabla 4-2:</b> Actualización de los dispositivos.....	42
<b>Tabla 5-2:</b> Análisis para la selección de hardware.....	43
<b>Tabla 6-2:</b> Función de los hilos del cableado.....	44
<b>Tabla 2-7:</b> Materiales y costo del Puntero Infrarrojo .....	50
<b>Tabla 1-3:</b> Datos a 45 grados. ....	91
<b>Tabla 2-3:</b> Datos a 75 grados. ....	91
<b>Tabla 3-3:</b> Datos a 90 grados. ....	92
<b>Tabla 4-3:</b> Características de las PDI en el mercado.....	93
<b>Tabla 5-3:</b> Materiales y costo de la Pizarra Interactiva.....	94
<b>Tabla 6-3:</b> Materiales y costo del Control Inalámbrico .....	94
<b>Tabla 7-3:</b> Costo total de la PDI propuesta .....	94
<b>Tabla 8-3:</b> Precios de PDI en el mercado.....	95
<b>Tabla 9-3:</b> Tiempo de respuesta de la PDI.....	97
<b>Tabla 10-3:</b> Funcionamiento de la PDI.....	98
<b>Tabla 11-3:</b> Precisión de la PDI.....	98
<b>Tabla 12-3:</b> Funcionamiento del Nunchuk.....	99
<b>Tabla 13-3:</b> Atención a la clase.....	100
<b>Tabla 14-3:</b> Participación en clase .....	100

## ÍNDICE DE GRAFICOS

<b>Gráfico 1-2:</b> Proceso de calibración.....	55
<b>Gráfico 2-2:</b> Proceso de funcionamiento general.....	55
<b>Gráfico 3-2:</b> Proceso de calibración.....	64
<b>Gráfico 1-3:</b> Cuadro comparativo de precios.....	95
<b>Gráfico 2-3:</b> Comparación entre dos PDI .....	96
<b>Gráfico 3-3:</b> Percepción de los Usuarios.....	97
<b>Gráfico 4-3:</b> Ubicación del cursor.....	98
<b>Gráfico 5-3:</b> Resultados del impacto en la clase. ....	100
<b>Gráfico 6-3:</b> Resultados del impacto en la clase. ....	101



## **ÍNDICE DE ANEXOS**

**ANEXO A:** Cálculo de ángulo de proyección

**ANEXO B:** Configuración de la función Call Library Function Node.

**ANEXO C:** Alimentación directa del Mando Wii.

**ANEXO D:** Encuesta

## INDICE DE ABREVIATURAS

<b>IPREX</b>	Instituto De Preparación De Exámenes
<b>IEEE</b>	Institute of Electrical and Electronics Engineers (Instituto de Ingeniería Eléctrica y Electrónica)
<b>IR</b>	Infrarrojo
<b>PDI</b>	Pantalla Digital Interactiva
<b>WLAN</b>	Wireless Lan Área Network
<b>PDIP</b>	Pantalla Digital Interactiva
<b>MEMS</b>	Sistemas Micro electromecánicos
<b>MOTS</b>	Multi-Object Tracking System
<b>GND</b>	Ground (Tierra)
<b>NI</b>	National Instruments
<b>GPIB</b>	Digital Parallel Communications Interface
<b>LabVIEW</b>	Laboratory Virtual Instrument Engineering Workbench
<b>PLCs</b>	Programmable Logic Controller
<b>USB</b>	Universal Serial Bus
<b>TCP</b>	Protocolo de Control de Transporte
<b>UDP</b>	User Datagram Protocol
<b>IrDA</b>	Infrared Wireless Communication
<b>OPC</b>	OLE for Process Control
<b>VI's</b>	Instrumentos Virtuales
<b>.NET</b>	Framework (Propietario de windows)
<b>CLR</b>	Common Language Runtime
<b>PInvoke</b>	Platform Invocation Services
<b>I2C</b>	Inter Integrated Circuits
<b>SDA</b>	Serial Data Line
<b>SCL</b>	Serial Clock Line
<b>ESPOCH</b>	Escuela Superior Politécnica de Chimborazo
<b>FIE</b>	Facultad de Informática y Electrónica
<b>IDE</b>	Integrated Development Environment
<b>VI</b>	Circuito Virtual
<b>Ms</b>	Milisegundos
<b>API</b>	Interfaz de programación de aplicaciones
<b>MAC</b>	Macintosh Operating System
<b>LabVIEW</b>	Laboratory Virtual Instrument Engineering Workbench

<b>PC:</b>	Computadora personal
<b>HID:</b>	Human Interface Device
<b>API:</b>	Interfaz De Programación De Aplicaciones
<b>TX:</b>	Transmisor
<b>RX:</b>	Receptor

## RESUMEN

Se analizó, diseñó e implementó una pantalla digital interactiva emulada utilizando tecnología infrarroja como un recurso didáctico para el Instituto de Preparación de Exámenes. Se realizó un análisis comparativo entre varias alternativas para la selección de un dispositivo con tecnología infrarroja idóneo y el software necesario para posteriormente ser utilizado en la implementación. El canal de comunicación inalámbrico entre los dispositivos utilizó principalmente la tecnología bluetooth definidos sobre el estándar del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE 802.2.15.1). En el diseño del sistema se requirió un Mando Wii (Wii Mote) que actuó como sensor infrarrojo captando la señal proveniente de un diodo IR y a su vez determinando su ubicación, el cual forma parte de un puntero infrarrojo; también se contó con el componente externo adicional Nunchuk que se utilizó para controlar el cursor inalámbricamente, además se necesitó el uso de una Arduino Nano el cual sirvió como un dispositivo intermediario para la recepción de la información enviada por el Nunchuk inalámbrico al ordenador. La aplicación se desarrolló en las plataformas Labview y Visual Studio C# integrándolas en un solo software encargado de procesar los datos recibidos por los componentes del sistema y ejecuta las instrucciones realizadas por el usuario a través del puntero. Los resultados obtenidos indicaron que la Pizarra Digital Interactiva Emulada posee un alto porcentaje (80%) de eficiencia y precisión en relación a una pantalla interactiva comercial, la cual también cuenta con un mando inalámbrico que facilita la interactividad entre la PDI y el usuario. Gracias al desarrollo de este prototipo se logró crear una PDI alternativa económica facilitando su adquisición y al mismo tiempo contribuyendo a la mejora de la didáctica y procesos de enseñanza/aprendizaje en las entidades educativas que buscan dinamizar e interactuar con los estudiantes para brindar un mejor servicio.

**Palabras clave:** <TECNOLOGIA Y CIENCIAS DE LA INGENIERÍA>, <TECNOLOGÍA DE COMUNICACIONES>, <PDI>, <INFOCUS>, <SOFTWARE>, <PANTALLA DIGITAL INTERACTIVA>, <RECURSO DIDACTICA>.

## SUMMARY

An emulated, interactive, digital screen was analyzed, designed and implemented utilizing infrared technology as a didactic resource for Exam Preparation Institution. A comparative analysis was performed between various alternatives to select an ideal device with infrared technology, as well as, the software necessary to implement said screen at a later date. The wireless communication channel between the devices made use primarily of the Bluetooth technology defined by the standard of the Electric and Electronic Engineers Institution (IEEE802.2.15.1). A Wii controller (Wii Mote) was necessary for the system's design. It acted as an infrared sensor by receiving the signal emitted from an IR diode which is part of an infrared pointer and simultaneously determining its location. Furthermore, the additional external component called a Nunchuk was used to control the cursor wirelessly. In addition, an Arduino Nano was needed and used as an intermediary device for the reception of the information sent by the wireless Nunchuk to the computer. The application was developed on the Labview and Visual Studio C# platforms which were integrates into one piece of software in charge of processing the data received by the system's components and executing the instructions given by the user through the pointer. The results indicated that the emulated, interactive, digital screen has a high percentage of efficiency (80%) and precision compared to an interactive commercial screen which also includes a wireless controller that facilitates the interaction between the Interactive Digital Screen (IDS) and the user. The development of this prototype offers a cheap alternative IDP which makes it easier to acquire and at the same time contributes to the improvement of didactics, as well as the teaching/learning processes in educational entities which want to motivate and interact with their students in order to offer a better service. We recommended utilizing a reflective surface such as white boards to increase the IDP's precision.

**Keywords:** <TECHNOLOGY AND ENGINEERING SCIENCES>, <COMUNICACION TECHNOLOGY>, <INTERACTIVE DIGITAL SCREEN (IDP) >, <INFOCUS PROJECTOR>, <LABVIEW (SOFTWARE) >, < VISUAL STUDIO (SOFTWARE)>, < DIDACTIC RESOURCE>.

## **INTRODUCCIÓN**

Este trabajo busca reducir el costo de las pantallas digitales interactivas realizando una emulación, obteniendo el mismo resultado de una PDI y adicional de bajo costo, en la actualidad las entidades educativas y/o academias buscan dinamizar e interactuar con los estudiantes para brindar un mejor servicio, al mismo tiempo la optimización de recursos es fundamental para la implementación de los mismos debido que las pantallas digitales interactivas tienen un elevado precio y hace que sea difícil su adquisición, el uso de nuevas tecnologías surge como respuesta para poder satisfacer en cierto grado los requerimientos de los usuarios.

El sistema de comunicación desarrollado básicamente constará de dos partes: La primera será crear la aplicación para la recepción y procesamiento de información proveniente del sensor infrarrojo y del acelerómetro, la segunda será establecer el canal de comunicación que se encargará de la transmisión y/o recepción de información entre los dispositivos antes mencionados.

Además se expone los objetivos, métodos y técnicas a seguir para el desarrollo y culminación de este trabajo de titulación.

## **ANTECEDENTES**

Los rayos infrarrojos son un tipo de radiación magnética, de mayor longitud de onda que la luz visible, pero menor que las microondas, es uno de los muchos tipos de luz que forman el espectro electromagnético. Isaac Newton realizó las primeras observaciones de los infrarrojos, pero el descubrimiento se le debe a William Herschel, quien en el año de 1800, colocó un termómetro de mercurio, en el espectro obtenido por un prisma de cristal, con el fin de medir el calor emitido por cada color. (DE LOPE, J. 2001, <http://www.dia.fi.upm.es/~jdlope/docs/delope01a.pdf>)

La tecnología infrarroja está basada en rayos luminosos que se mueven en el espectro infrarrojo, permite la comunicación entre dos modos, usando una serie de leds infrarrojos para ello. Los estándares IrDa soportan una amplia gama de dispositivos eléctricos, informáticos y de comunicaciones, permite la comunicación bidireccional entre dos extremos a velocidades que oscilan entre los 9.600 bps y los 4 Mbps. (Carrión. J, & Torres. R, 2012, <http://repositorio.espe.edu.ec/bitstream/21000/5442/1/T-ESPE-033352.pdf>)

Esta tecnología se encontraba en muchos ordenadores portátiles y teléfonos móviles de finales de los 90's y principios de la década del 2000, pero fue gradualmente desplazada por tecnologías como WI-FI y Bluetooth. Sin embargo, así mismo surgen nuevas tendencias y el uso de la tecnología infrarroja sigue en marcha, por sus prestaciones en redes de área local inalámbricas (WLAN's) y su bajo costo. (CARRIÓN. J, & TORRES. R, 2012, <http://repositorio.espe.edu.ec/bitstream/21000/5442/1/T-ESPE-033352.pdf>)

Actualmente, los infrarrojos se utilizan especialmente por sus características de emisión y recepción de comandos a distancia, comunicación a corta distancia entre periféricos y computadores, lo que proporciona múltiples aplicaciones en la medicina, actividades policiales y de seguridad, aplicaciones militares, sistemas electrónicos, entre otros. Sin embargo cabe recalcar que dicha tecnología tuvo un gran desarrollo y evolución durante muchos años llegando a formar parte de diversos dispositivos beneficiándose de sus características de transmisión para poder compartir archivos, imágenes, audio, etc. (PARUELO. J, 2008, <http://revistaecosistemas.net/index.php/ecosistemas/article/view/83> )

En la mayoría de las instituciones educativas de nuestro país ya sean públicas o privadas, la enseñanza ha sido una labor complicada, debido a la falta de creatividad y dinamismo, como es el caso de las pizarras convencionales que se utilizaban unos años atrás lo que genera una gran problemática al momento de impartir los contenidos didácticamente, logrando que la materia se vuelva monótona y hasta cierto punto tediosa. (CARRIÓN. J, & TORRES. R, 2012, <http://repositorio.espe.edu.ec/bitstream/21000/5442/1/T-ESPE-033352.pdf>)

Las pizarras interactivas están revolucionando la didáctica al momento de impartir los contenidos más aun ya que en este proyecto se pretende desarrollar mediante la tecnología infrarroja que ofrece un amplio ancho de banda para la transmisión a muy altas velocidades, además de utilizar un protocolo simple y componentes sumamente económicos y de bajo consumo de potencia. (ASENCIO, G et al, 2011, [http://apps.elsevier.es/watermark/ctl\\_servlet?\\_f=10&pidet\\_articulo=90029254&pidetusuario=0&pcontactid=&pidet\\_revista=331&ty=148&accion=L&origen=zonadelectura&web=www.elsevier.es&lan=es&fichero=331v08n04a90029254pdf001.pdf](http://apps.elsevier.es/watermark/ctl_servlet?_f=10&pidet_articulo=90029254&pidetusuario=0&pcontactid=&pidet_revista=331&ty=148&accion=L&origen=zonadelectura&web=www.elsevier.es&lan=es&fichero=331v08n04a90029254pdf001.pdf))

En una sociedad globalizada que continuamente se va adaptando al desarrollo tecnológico surgen nuevos planteamientos educativos en contenidos, recursos tecnológicos actuales o nuevas metodologías de enseñanza, etc. En este contexto, las nuevas tecnologías de información y comunicación ponen a la disposición las pantallas interactivas como las pizarras digitales, táctiles, etc., herramientas basadas en tecnología infrarroja, capacitiva, resistiva, entre otras. (DE LOPE, J. 2001, <http://www.dia.fi.upm.es/~jdlope/docs/delope01a.pdf>)

IPREX es un centro de preparación y nivelación académica previo al ingreso de Politécnicas y Universidades del País, con el compromiso y convicción de servicio a la sociedad. Cuenta con tecnologías actuales, red de datos disponible en la mayoría del edificio y una infraestructura adecuada para sus servicios.

El desarrollo del presente proyecto tiene como referencia los estudios realizados en la Universidad Complutense de Madrid, realizado por Valentín Blanco Gómez & Raúl Sampedro Ruíz en el 2014 con el tema “Diseño e implementación de pizarras digitales de bajo coste”; y en nuestro país el estudio desarrollado en la Universidad de las Fuerzas Armadas (anteriormente conocida como Escuela Politécnica del Ejercito) por Jonathan Carrión Castellanos & Roberto Torres Fonseca, R. D. en el año 2012 con el tema “Implementación de una pizarra digital interactiva multiusuario con desarrollo de aplicación para Tele Educación en personas con capacidades diferentes”. (CARRIÓN. J, & TORRES. R, 2012, <http://repositorio.espe.edu.ec/bitstream/21000/5442/1/T-ESPE-033352.pdf>)

## **FORMULACIÓN DEL PROBLEMA**

¿Será posible llevar a los estudiantes de la academia IPREX a una educación basada en un entorno digital de forma entretenida e interactiva, invirtiendo la menor cantidad de recursos económicos?

## **SISTEMATIZACIÓN DEL PROBLEMA**

¿Cuál será el tiempo de respuesta de la comunicación entre el dispositivo infrarrojo y el ordenador?

¿Cuál será el nivel de efectividad que tendrá la interfaz con respecto a las peticiones del usuario?

¿Cuál será a tasa de transferencia que utilice la tecnología infrarroja para enviar las peticiones de usuario hacia el Mando Wii?

¿Cuáles serán los parámetros necesarios para obtener un adecuado funcionamiento de la PDI emulada?

¿Cuál sería el impacto producido mediante el uso de la emulación de una pantalla digital interactiva en las tareas de enseñanza de IPREX?



## JUSTIFICACIÓN TEÓRICA

En la actualidad las entidades educativas y/o academias buscan dinamizar e interactuar con los estudiantes para brindar un mejor servicio, al mismo tiempo la optimización de recursos es fundamental para la implementación de los mismos debido que las pantallas digitales interactivas tienen un elevado precio y hace que sea difícil su adquisición, el uso de nuevas tecnologías surge como respuesta para poder satisfacer en cierto grado los requerimientos de los usuarios, una de ellas es la tecnología Infrarroja la cual nos permite detectar movimiento de los objetos, transmitir y recibir datos de forma inmediata mediante ondas electromagnéticas entre dispositivos que cuenten con ésta tecnología.

Para el desarrollo del proyecto se empleara un plataformas de diseño de software, la misma que se utilizara para la captura y análisis de la información y la ejecución de comandos e instrucciones entre el sistema infrarrojo y el ordenador mediante una conexión inalámbrica, logrando de esta manera interpretar las acciones del usuario sobre una superficie plana (pantalla, pizarra, etc.) y convertirlas en un lenguaje que entienda el ordenador.

Esta plataforma de programación puede estar basado en software de código abierto como lenguaje C o Python que son lenguajes de programación de alto nivel, también en un software licenciado como Labview que es un lenguaje de desarrollo con una sintaxis de programación gráfica que facilita visualizar, crear y codificar sistemas de ingeniería, estas opciones se analizaran minuciosamente en el desarrollo del trabajo de titulación.

Mediante la tecnología infrarroja y el lenguaje de programación que nos brinde las mejores prestaciones, se buscara reducir el tiempo de respuesta que se genera por cada instrucción del usuario, el mismo que resulta muy inconveniente y notorio al ser utilizado, creando un entorno más interactivo, puesto que la tecnología infrarroja presta las siguientes características:

- Ofrece una amplio ancho de banda que transmite señales a velocidades muy altas (alcanza los 10 Mbps).
- Tiene una longitud de onda cercana a la de la luz y se comporta como ésta (no puede atravesar objetos sólidos como paredes, por lo que es inherentemente seguro contra receptores no deseados).
- Utiliza un protocolo simple y componentes sumamente económicos y de bajo consumo de potencia (Camello. P, 2012, <http://baezjacobotics.blogspot.com/2012/10/wifi-bluetooth-infrarrojo.html>).

## **JUSTIFICACIÓN APLICATIVA**

En la actualidad existen proyectores instalados en las aulas de clase, auditorios, etc. Los cuales nos muestran una pantalla común y poco didáctica, la pantalla digital interactiva emulada nos brinda una interfaz con el usuario y motiva el deseo de aprender de los estudiantes, mejora la comprensión, aumenta la atención y retentiva debido a que los estudiantes participan más, facilita el debate, etc. El ponente puede concentrarse más en observar a los alumnos y contestar sus dudas porque no debe estar mirando todo el tiempo la pantalla del computador.

Es evidente que gracias a la implementación de la PDI emulada ara el proceso de enseñanza más didáctico e intuitivo mejorando el entorno de aprendizaje. La facilidad de aprendizaje es una medida del tiempo requerido para trabajar con cierto grado de eficiencia en el uso de la herramienta, y alcanzar un cierto grado de retención de estos conocimientos luego de cierto tiempo de no usar la herramienta o sistema.

El desarrollo del proyecto se realizará mediante la implementación de un prototipo en las aulas de capacitación de IPREX, convirtiendo una pantalla común en un entorno interactivo, este recurso didáctico cuenta con un control inalámbrico que utiliza la tecnología bluetooth, esto crea un entorno más amigable y dinámico ya que los alumnos también podrían participar en la clase y no necesariamente deben acercarse a la pantalla, también cuenta con un puntero de luz infrarroja, esto le permite al usuario tener un contacto directo con la pantalla teniendo múltiples beneficios.

El sistema propuesto contara con los siguientes dispositivos:

- Mando Wii
- Controlador inalámbrico de la PDI emulada.
- Adaptador Bluetooth.
- Puntero de luz infrarroja.

## **OBJETIVOS**

### **Objetivos Generales**

Diseñar e implementar una pantalla digital interactiva emulada utilizando tecnología infrarroja como un recurso didáctico para IPREX

### **Objetivos Específicos**

- Estudiar el estado del arte de la tecnología infrarroja.
- Diseñar un sistema que permita interpretar la comunicación entre el dispositivo infrarrojo y el ordenador mediante un lenguaje de programación para diseño y desarrollo de aplicaciones.
- Diseñar e implementar un control inalámbrico para complementar el sistema de mando de la PDI emulada.
- Verificar el adecuado funcionamiento de la emulación de la pantalla digital tomando como referencia las PDI existentes en el mercado.

## CAPITULO I

### 1 MARCO REFERENCIAL

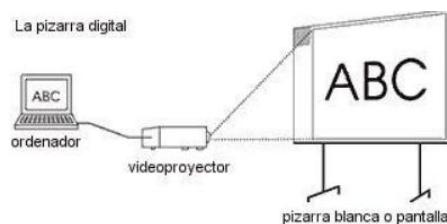
#### 1.1 Pantalla Digital Interactiva

Podemos definir Pizarra Digital Interactiva como un sistema tecnológico, generalmente integrado por un ordenador, un proyector y un dispositivo de control de puntero, esto permite proyectar en una superficie contenidos digitales en un formato idóneo para visualización en grupo, convirtiéndole en un entorno más amigable e interactivo. La PDI ofrece al docente acostumbrado a las pizarras tradicionales de tiza o marcador, encontrar un recurso muy cercano a la tradición pedagógica que incorpora las TIC en el aula de manera visible y transparente.

Todos los alumnos pueden ver y actuar con los equipos informáticos, individual o grupalmente ante todos sus compañeros. La PDI supera la sensación de “caja negra” y hace posible una amplia variedad de especificaciones y capacidades. (Gallego. M et al. 2009, [http://campus.usal.es/~teoriaeducacion/rev\\_numero\\_10\\_02/n10\\_02\\_gallego\\_cacheiro\\_dulac.pdf](http://campus.usal.es/~teoriaeducacion/rev_numero_10_02/n10_02_gallego_cacheiro_dulac.pdf))

A continuación los beneficios que ofrece:

- La manipulación fácil y rápida de textos e imágenes
- Utilizar la Web y sus recursos ante toda la clase
- Utilizar y demostrar diferentes tipos de software
- Escribir y resaltar los aspectos de interés sobre textos, imágenes o vídeos, etc.



**Figura 1-1:** Diseño de la PDI

**Fuente:**<http://www.ardilladigital.com/DOCUMENTOS/TECNOLOGIA%20EDUCATIVA/TICs/T9%20PIZARRA%20DIGITAL/09%20LA%20PIZARRA%20DIGITAL.pdf>

### 1.1.1 Pizarra Digital Interactiva Portátil (PDIP)

Estas pantallas son fijas por lo que tiene dificultad al momento de querer movilizarla. Es una pantalla digital interactiva proyectada por un infocus en una superficie, y un accesorio que funciona en conjunto con un control de puntero convirtiendo así la superficie en una PDI. Este dispositivo tecnológico es más económico que las PDI fijas.



**Figura 2-1:** PDI con dispositivos de control.

**Fuente:** <http://www.ardilladigital.com/DOCUMENTOS/TECNOLOGIA%20EDUCATIVA/TICs/T9%20PIZARRA%20DIGITAL/09%20LA%20PIZARRA%20DIGITAL.pdf>

Es un sustituto de las PDI, envían la información al ordenador de forma inalámbrica tomando así el control del mismo y realizando cualquier actividad como si estuviese directamente en el PC.



### Figura 3-1: Pizarras MIMO

Fuente: <http://www.ardilladigital.com/DOCUMENTOS/TECNOLOGIA%20EDUCATIVA/TICs/T9%20PIZARRA%20DIGITAL/09%20LA%20PIZARRA%20DIGITAL.pdf>

#### 1.1.2 Pizarras De Pantalla Frontal

En esta pantalla tenemos la posibilidad de escribir con el dedo o con rotuladores especiales, la desventaja es que puede ser propenso a daños debido que si escribimos con un rotulador diferente al recomendado o si recibe golpes, las sombras también es una desventaja porque la imagen se proyecta frontalmente.

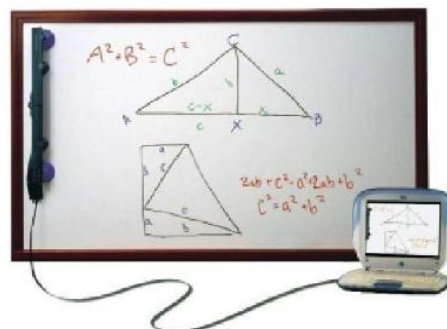


Figura 4-1: Pantalla digital frontal o retroalimentada.

Fuente: <http://capacitacionpizarradigitalinteractiva.blogspot.com/2014/11/pizarra-digital-interactiva-pdi.html>

#### 1.1.3 Pizarra de Barras de Coordenadas o Ángulos

Para visualizar se utiliza cualquier superficie plana, esta barra proporciona un sistema de coordenadas las cuales envía directamente al ordenador de forma. La desventaja es que no podemos escribir sobre esta PDI con el dedo.



### **Figura 5-1:** Pantalla de barra de ángulos

**Fuente:**<http://www.ardilladigital.com/DOCUMENTOS/TECNOLOGIA%20EDUCATIVA/TICs/T9%20PIZARRA%20DIGITAL/09%20LA%20PIZARRA%20DIGITAL.pdf>

#### **1.1.4 Pantallas Planas Táctiles**

Esto presenta una ventaja cuando cuentas con una pantalla LCD o de plasma porque únicamente debemos colocar el dispositivo interactivo en la parte inferior el cual la convierte en una PDI.



**Figura 6-1:** Pantalla LCD, Plasma interactivas

**Fuente:**<http://www.ardilladigital.com/DOCUMENTOS/TECNOLOGIA%20EDUCATIVA/TICs/T9%20PIZARRA%20DIGITAL/09%20LA%20PIZARRA%20DIGITAL.pdf>

#### **1.1.5 Tecnologías de las PDI**

##### **1.1.5.1 Electromagnética**

En este tipo de tecnología se utiliza un lápiz especial, el mismo que actúa como puntero y como superficie se utiliza una malla la cual detecta la señal del puntero al momento que toca la superficie, esta detección del campo electromagnético emitido por el puntero permite la localización del punto señalado, esta pizarra cuenta con muy alta precisión. (Fernández, R. 2013, pp 7-8).

##### **1.1.5.2 Resistiva**

La pizarra está formado por dos capas separadas, la exterior es deformable al tacto. La presión aplicada facilita el contacto al unirse las dos láminas, provocando una variación de la resistencia eléctrica permitiendo determinar la ubicación del punto. (Fernández, R. 2013, <http://revistas.unica.cu/index.php/uciencia/article/view/1/1>).

#### *1.1.5.3 Óptica*

La superficie de la pizarra está cubierta por barras de luz infrarroja y en la parte superior en cada lado se encuentran dos cámaras infrarrojas mismas que están monitoreando que no se rompa la línea de vista entre las cámaras y las barras infrarrojas, para localizar la ubicación del punto x, y se debe presionar en alguna zona de la pantalla y se calcula el área donde se ha distorsionado la señal. (Fernández, R. 2013, <http://revistas.unica.cu/index.php/uciencia/article/view/1/1>).

#### *1.1.5.4 Infrarroja*

El puntero emite una señal infrarroja pura al entrar en contacto con la pantalla, el receptor se ubica a una distancia determinada el cual traduce la ubicación del punto (o los puntos) infrarrojos a coordenadas cartesianas, las que son usadas para ubicar el mouse.

La ventaja de utilizar tecnología es que no depende de sensores adicionales, ni soportes o superficies sensibles ya que se puede proyectar en cualquier superficie. Además no está limitado a un área de proyección pudiendo incluso expandirse a varios metros cuadrados. (Fernández, R. 2013, <http://revistas.unica.cu/index.php/uciencia/article/view/1/1>).

### ***1.1.6 Elementos que integran la PDI***

Una pizarra interactiva debe incluir como mínimo los siguientes elementos:

- Ordenador o Pc, debe contar con los elementos básicos y ser capaz de reproducir toda la información multimedia almacenada de cualquier tipo o extensión. El sistema operativo instalado en el ordenador tiene que ser compatible con la aplicación de la pizarra.
- Proyector, necesario para la visualización de la imagen del ordenador sobre la pizarra, se debe considerar una buena luminosidad y resolución. La ubicación del proyector depende de la imagen que se desea obtener a una distancia que provea una buena resolución.
- Medio de conexión, canal a través del cual se comunican el ordenador y la pizarra, Existen algunos tipos de conexiones entre ellos bluetooth, cable (USB, paralelo) o conexiones basadas en tecnologías de identificación por radiofrecuencia.



- Pantalla, superficie sobre la que se proyecta la imagen del ordenador mediante el infocus y que se controla mediante un puntero o incluso con el dedo según la tecnología.
- Software de la PDI, proporcionada por el fabricante o distribuidor, cuya función principal permite la gestión de la pizarra, captura de imágenes y pantallas, dispone de plantillas de diversos recursos y herramientas educativas, procesos internos que permiten la interactividad, entre otras. (Fernández, R. et al, 2012, <http://revistas.unica.cu/index.php/uciencia/article/view/1/1>).

## 1.2 Wiimote

El éxito del mando Wii es porque está basado en el reconocimiento del movimiento, utiliza tecnología giroscópica que permite detectar el movimiento que realizamos con el mando por completo, además de la velocidad con que lo movemos. El mando, de forma alargada, es de un tamaño muy reducido y cuenta con ocho botones además de una cruceta, aunque durante los juegos el uso se limita casi por completo a dos de ellos, siendo los otros para realizar funciones concretas, incluso uno de ellos se utiliza en exclusiva para encender y apagar la consola a distancia. (Valero. R, 2010, [http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes\\_RodrigoValeroR\\_Integraci%C3%B3nMandoInalambrico\\_2010.pdf?sequence=1](http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes_RodrigoValeroR_Integraci%C3%B3nMandoInalambrico_2010.pdf?sequence=1)).



**Figura 7-1:** Mando Wii Remote Plus

**Fuente:** <http://wiimovement.com/wii-motion-remote-compared-to-the-wii-motion-plus-remote/>

El mando es totalmente inalámbrico y permite apuntar a la pantalla en ciertos títulos, al igual que ciertos controladores con forma de pistola. La alimentación del mando se realiza mediante dos pilas de las que se indica la carga mediante las luces que dispone el mando en la parte inferior, las cuales sirven también para indicar nuestro número de jugador. Ni la detección del movimiento, ni el hecho de ser un mando inalámbrico son problemas para que este pequeño controlador incluya vibración, y no sólo eso, sino que también cuenta con un

pequeño altavoz que puede reproducir algunos sonidos o frases en momentos claves del juego, para conseguir una involucración mayor por parte del jugador.

Aunque este mando es el corazón de Wii, no es la única forma de control en los juegos, ya que al mando principal Wii Remote podemos conectar una extensión conocida como “nunchuk”, la cual va unida por un cable, de forma que con una mano manejamos el mando de Wii y con la otra el nunchaku. El nunchaku cuenta con dos gatillos y un joystick analógico, aunque también permite detectar cierto movimiento, en concreto movimientos rápidos que realicemos con él. (Valero. R, 2010, [http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes\\_RodrigoValeroR\\_Integraci%C3%B3nMandoInalambrico\\_2010.pdf?sequence=1](http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes_RodrigoValeroR_Integraci%C3%B3nMandoInalambrico_2010.pdf?sequence=1))

### ***1.2.1 Funcionamiento del mando Wii***

Dispone de un filtro infrarrojo delante de la cámara de alta resolución, con lo que la cámara solo recibe la imagen de fuentes emisoras de infrarrojos. Si situamos un led infrarrojo delante, esta cámara percibe un punto y envía su posición X e Y, respecto la imagen que capta la cámara en ese momento, tiene una resolución de 1024 píxeles de ancho por 768 de alto. Por tanto, el punto superior izquierdo que ve la cámara sería la posición (0,0), el punto superior derecho la posición (0,1024), el punto inferior izquierdo la posición (768,0) y el punto inferior derecho la posición (1024,768). El mando dispone de un chip que posiciona el punto que recibe la cámara sobre el plano que capta, y envía la posición del punto por Bluetooth al dispositivo que esté conectado. Este sistema de posicionamiento se llama “Sistema de posicionamiento multiobjeto”. (Samaniego. R, 2009, [http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes\\_RodrigoValeroR\\_Integraci%C3%B3nMandoInalambrico\\_2010.pdf?sequence=1](http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes_RodrigoValeroR_Integraci%C3%B3nMandoInalambrico_2010.pdf?sequence=1))

Se trata de un sistema de procesamiento de imagen asistido por ordenador, que ya se encuentra integrado en el hardware, lo que permite localizar la posición de los puntos brillantes que recibe la cámara. El chip de procesamiento de la posición de los puntos puede procesar hasta 4 puntos diferentes simultáneamente, y enviar la posición de todos ellos. Por tanto, podríamos posicionar 4 punteros infrarrojos diferentes si se deseara. Si hubiera más de 4 punteros en la superficie, escogería los 4 puntos más brillantes por defecto. (Samaniego. R, 2009, [http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes\\_RodrigoValeroR\\_Integraci%C3%B3nMandoInalambrico\\_2010.pdf?sequence=1](http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes_RodrigoValeroR_Integraci%C3%B3nMandoInalambrico_2010.pdf?sequence=1)).

Cada vez que el mando Wii envía la nueva ubicación del punto infrarrojo debe esperar 100 ms, este tiempo se conoce como tiempo de muestreo. “También dispone de una pequeña memoria

de 4 KB utilizada para guardar el emoticono del jugador que utiliza el mando en ese momento” (Samaniego, R, 2009, [http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes\\_RodrigoValeroR\\_Integraci%C3%B3nMandoInalambrico\\_2010.pdf?sequence=1](http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes_RodrigoValeroR_Integraci%C3%B3nMandoInalambrico_2010.pdf?sequence=1))

### 1.2.2 Direcciones de memoria que recibe el mando Wii

**Tabla 1-1:** Direcciones de memoria del Mando Wii

Dirección	Función
0x11	Indicación del número de jugador y activación/desactivación del vibrador.
0x12	Indicación de la ID del mando
0x13	Activación del sensor IR
0x14	Activación del altavoz
0x15	Estado del mando
0x16	Escribir datos en memoria
0x17	Leer datos de memoria
0x18	Datos enviados al altavoz
0x19	Silenciar altavoz
0x1a	Activar sensor IR 2

**Fuente:** (Ricardo Bonache Samaniego, 2009)

#### 1.2.2.1 Puertos y estados

**Tabla 2-1:** Puertos y estados del Mando Wii

Dirección	Función
0x20	Puerto de expansión
0x21	Lectura de datos de memoria
0x22	Escritura de datos de memoria
0x30	Botones
0x31	Estado de botones/Información acelerómetro
0x32	Estado de botones/Información sensor IR
0x33	Estado de botones/Información acelerómetro
0x34	Estado de botones/Información sensor IR
0x35	Estado de botones/Información acelerómetro
0x36	Estado de botones/Información sensor IR
0x37	Estado de botones/Información acelerómetro

0x3d	Estado de botones/Información sensor IR
0x3e	Estado de botones/Información acelerómetro
0x3f	Estado de botones/Información sensor IR

**Fuente:** (Bonache Samaniego, Ricardo. 2009)

La información sobre el puerto de expansión indica si hay algún elemento auxiliar conectado al puerto que hay debajo del mando. Las direcciones de lectura y escritura de datos en la memoria se utilizan para conocer qué datos han sido los últimos que se han introducido, tanto en lectura como en escritura. (Samaniego. R, 2009, [http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes\\_RodrigoValeroR\\_Integraci%C3%B3nMandoInalambrico\\_2010.pdf?sequence=1](http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes_RodrigoValeroR_Integraci%C3%B3nMandoInalambrico_2010.pdf?sequence=1)).

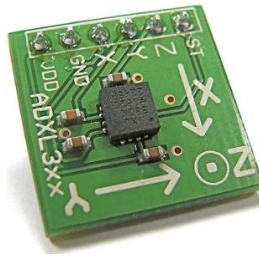
### ***1.2.3 Componentes internos del mando Wii***

#### ***1.2.3.1 Acelerómetro***

Es aquel que permite realizar giros e inclinaciones, además nos proporciona los datos de los movimientos que realiza el mando actualizándose periódicamente, también se los conoce a los acelerómetros como MEMS. El chip utilizado en el Wii mote proviene de Analog Devices y es exactamente el modelo ADLXL330. Este chip está diseñado para tener un consumo muy bajo, alargando la vida de las pilas. Su rango de detección es de  $\pm 3,6$  gravedades y una resolución de 300 mV/g. (Samaniego. R, 2009, [http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes\\_RodrigoValeroR\\_Integraci%C3%B3nMandoInalambrico\\_2010.pdf?sequence=1](http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes_RodrigoValeroR_Integraci%C3%B3nMandoInalambrico_2010.pdf?sequence=1))

Estudiando el funcionamiento de estos chips, se descubre que miden la aceleración. Para comprender, imagine una pila con dos minúsculas placas dentro del sensor. Una se mantiene fija, pero la otra se mueve. Los electrones que rodean la placa que se ha movido varían, y midiendo la capacitancia, el sensor es capaz de detectar el movimiento realizado.

El dispositivo puede así enviar datos de movimientos relativos en todos los ejes de coordenadas, aparte de los datos de aceleración, detectando (dentro de sus limitaciones) giros, movimientos en el aire, inclinaciones. Estos datos se envían al chip Broadcom, encargado de la emisión de datos a través de Bluetooth. (Samaniego. R, 2009)



**Figura 8-1: Acelerómetro**

**Fuente:** (<https://www.flickr.com/photos/oskay/1193147290/>)

#### 1.2.3.2 Cámara Infrarroja

El mando Wii tiene incorporado en su parte superior una cámara monocroma con una resolución de 128x96, con procesado de imagen mediante hardware ya incorporado, este es capaz de posicionar hasta 4 puntos infrarrojos en movimiento. En la parte delantera del mando encontramos un plástico de color negro, se trata de un filtro que deja pasar rayos infrarrojos, de esta forma la cámara únicamente podrá ver rayos o puntos de un emisor infrarrojo. Los datos obtenidos de la posición de los puntos infrarrojos se enviar por medio de la tecnología Bluetooth al dispositivo que esté conectado el mando como por ejemplo un ordenador, una consola de Nintendo, etc. El procesado por hardware únicamente nos permite enviar información de los puntos infrarrojos mas no de imágenes debido que no recibe toda lo que capta la cámara.

Los ángulos con los que opera la cámara es de 33 grados en el eje horizontal y 23 grados en el eje vertical, además puede detectar un haz de luz IR hasta una distancia de 10 pies, Con el filtro de infrarrojos intacto, los diodos que tienen una longitud de onda de 940 nm son vistos fácilmente por el sensor, por el contrario los que tienen longitud de onda de 850 nm la desventaja es que la distancia entre el diodo y la cámara es pequeña y al estar cerca la intensidad de luz es fuerte lo que dificulta ubicar el punto correctamente.

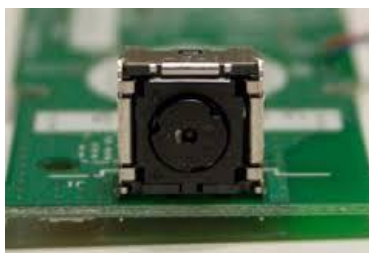
En caso de retirar el filtro de infrarrojos del mando, podríamos posicionar cualquier punto brillante. El procesador MOTS de la marca PixArt incorporado utiliza un análisis de subpíxeles, los cuales multiplica automáticamente por 8, llegando a obtener una resolución de 1024x768 para los puntos posicionados.

La cámara de infrarrojos se activa poniendo el bit 2 en el reporte de salida como 0x13 y 0x1a:

(a2) 13 04

(a2) 1a 04

La primera permite una frecuencia de píxel de 24 MHz en el pin 7 de la cámara. La segunda en el pin 4.



**Figura 9-1:** Cámara del mando Wii

**Fuente:** <http://deeea.urv.cat/public/PROPOSTES/pub/pdf/1231pub.pdf>



**Figura 10-1:** Ubicación de la cámara del mando Wii

**Fuente:** <http://www.mundopepone.com/2012/12/06/como-functiona-el-mando-de-wii/>

#### 1.2.3.2.1 *Características*

La cámara IR tiene un campo de visión efectivo de unos 33 grados en horizontal y 23 grados en vertical. Con el filtro de paso IR, las fuentes de 940nm se detectan con aproximadamente el doble de la intensidad a las fuentes equivalentes de 850nm, pero no se resuelven, así a corta distancia. Si se quita el filtro, se puede rastrear cualquier objeto brillante. Sin embargo, el filtro IR no sólo hace referencia la ventana de plástico oscuro del mando sino también una losa teensy de vidrio dicroico revestido en el interior del módulo de la cámara.

#### 1.2.3.2.2 *Activación de la cámara*

Debemos tener en cuenta se debe establecer el bit 2 es decir (0x04) en el primer byte en la salida a los registros q se va escribir.

Ahora, para encender la cámara debemos realizar el siguiente procedimiento:

1. Habilitar el byte bajo del registro de activación de la cámara IR, enviando 0x04 al registro de salida 0x03.

2. Habilitar el byte alto del registro de activación de la cámara IR, enviando 0x04 al registro de salida 0x1a.
3. Escribir 0x08 en el registro 0xb00030.
4. Escribir bloque 1 de información de sensibilidad de la cámara en la dirección 0xb00000.
5. Escribir bloque 2 de la información de sensibilidad de la cámara en la dirección 0xb0001a.
6. Escribir el número de modo en el registro 0xb00033.
7. Escribir 0x08 en el registro 0xb00030 nuevamente.

Después de realizar este procedimiento la cámara se encuentra en uno de los tres estados:

- La cámara infrarroja se encuentra encendida pero no recibe ningún dato
- La cámara infrarroja se encuentra encendida y está recibiendo los datos
- La cámara infrarroja se encuentra tomando los datos a la máxima sensibilidad.

Para evitar el estado aleatorio se debe configurar unos 50 ms entre cada byte que se transmite y para seleccionar el estado que se necesita se debe repetir el procedimiento. Para obtener una mejor precisión debemos ajustar la sensibilidad al máximo.

#### 1.2.3.2.3 Configuración de sensibilidad

La sensibilidad se puede controlar gracias a dos bloques de configuración cuya longitud del primer bloque es 9 bytes y del segundo 2 bytes, con los siguientes datos podemos configurar la sensibilidad de acuerdo a nuestros requerimientos.

**Tabla 3-1:** Registros de Sensibilidad

Bloque 1	Bloque 2	Nivel de sensibilidad
00 00 00 00 00 00 FF 00 0C	00 00	Máxima sensibilidad
00 00 00 00 00 00 90 00 41	40 00	Alta sensibilidad
02 00 00 71 01 00 64 00 fe	fd 05	1
02 00 00 71 01 00 96 00 b4	b3 04	2
02 00 00 71 01 00 aa 00 64	63 03	3
02 00 00 71 01 00 c8 00 36	35 03	4
07 00 00 71 01 00 72 00 20	1f 03	5

**Fuente:** Ricardo Bonache Samaniego, 2009

El último byte del Bloque 1 determina intensidad de la sensibilidad intensidad. Los dos bytes del segundo bloque deben ser cero para que el rango de sensibilidad esté disponible en su totalidad. Cuando el ultimo byte del bloque 1 es 0x0C y el segundo byte del bloque 2 es 0x00 obtenemos la mayor sensibilidad. Es decir a menor luz incrementa la precisión al momento de detectar un punto de infrarrojo debido que cualquier punto brillante será descartado. Si escogemos el nivel 1 de sensibilidad también estamos reduciendo la resolución llegando así a la resolución del sensor que es de 128x96.

#### *1.2.3.2.4 Formato de los datos*

La cámara de infrarrojos puede devolver diferentes conjuntos de datos que describen los objetos que está rastreando. Cuando la cámara IR identifica un objeto, se asigna a la primera posición del vector disponible. Si un objeto se mueve fuera de la vista, si el punto se mueve fuera de la zona de la cámara la posición de memoria se asigna como vacía (devuelve datos 0xFF), los otros puntos que la cámara está visualizando mantienen su posición en memoria. Por ejemplo, si la cámara está visualizando y si el primer punto sale de la zona que cubre la cámara, los datos envía serán [vacío, segundo punto xy, vacío, vacío]. Con más de cuatro puntos visibles, la cámara es propenso a equivocarse. Esto podría permitir que la percepción de más de cuatro objetos tenga baja fiabilidad y velocidad de respuesta. Los datos que nos puede enviar el mando Wii de los puntos son de tres tipos: Básico, Extendido o Completo.

##### *1.2.3.2.4.1 Modo Básico*

En este modo, la cámara infrarroja proporciona datos cuya longitud son 10 bytes esto correspondiente a las ubicaciones X e Y de cada uno de los cuatro puntos. Cada ubicación se codifica en 10 bits y tiene un rango de 0 a 1023 para la coordenada X, y 0 a 767 para la coordenada Y. Cada par de puntos se empaqueta en 5 bytes. El formato que se sigue para enviar la ubicación de los puntos es:



	Bit							
Byte	7	6	5	4	3	2	1	0
0	X1<7:0>							
1	Y1<7:0>							
2	Y1<9:8>	X1<9:8>		Y2<9:8>		X2<9:8>		
3	X2<7:0>							
4	Y2<7:0>							

**Figura 11-1:** Formato de datos en modo básico

**Fuente:** (Ricardo Bonache Samaniego, 2009, p.23)

#### 1.2.3.2.4.2 *Modo Extendido*

En este modo, la cámara de infrarrojos devuelve los mismos datos que lo hace en modo básico, lo que añade es un valor de tamaño aproximado para cada punto. Los datos se devuelven como 12 bytes, tres bytes por objeto. Tamaño de los datos del punto identificado por la cámara tiene un rango de 0 a 15. El formato de datos para este modo es:

Byte	Bit							
	7	6	5	4	3	2	1	0
0	X<7:0>							
1	Y<7:0>							
2	Y<9:8>		X<9:8>		S<3:0>			

**Figura 12-1:** Formato de datos en modo extendido.

**Fuente:** (Ricardo Bonache Samaniego, 2009, p.23)

#### 1.2.3.2.4.3 *Modo Completo*

En este modo, la cámara de infrarrojos arroja más los datos, Envía 9 bytes por cada punto detectado, tomando el ejemplo de los modos anteriores, si enviamos los datos de 4 puntos en este caso sería un total de 36 bytes. Para facilitar el envío los datos se divide en dos tramas de 18 bytes cada una posteriormente se envían a dos registros de entrada (registros 0x3e / 0x3f respectivamente). Los tres primeros bytes de cada punto son el iguales al modo extendido, los siguientes cuatro bytes corresponden a la posición del punto es decir el valor máximo y mínimo X e Y el siguiente byte es un 0 y es utilizado para separar los datos, el ultimo byte nos indica la intensidad luminosa que tiene el punto. El formato de datos es el siguiente:

	Bit							
Byte	7	6	5	4	3	2	1	0
0	X<7:0>							
1	Y<7:0>							
2	Y<9:8>		X<9:8>		S<3:0>			
3	0	X min<6:0>						
4	0	Y min<6:0>						
5	0	X max<6:0>						
6	0	Y max<6:0>						
7	0							
8	Intensidad<7:0>							

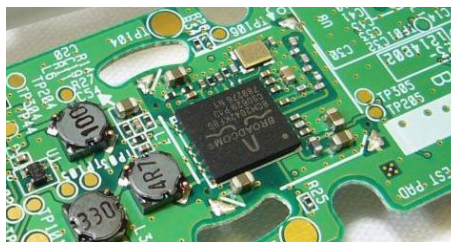
**Figura 13-1:** Formato de datos en modo completo

**Fuente:** (Ricardo Bonache Samaniego, 2009, p.24)

Además el Wii mote cuenta con leds indicadores y un parlante para crear un ambiente más real.

#### 1.2.3.3 *Chip emisor/receptor Bluetooth*

Para establecer comunicación entre el mando y la consola Wiimote dispone de un chip emisor/receptor Bluetooth, la velocidad de transferencia es de 2.1 Mbits/s. la empresa Broadcom Technologies es la que provee el chip, en este caso se usa el modelo BCM2042. (Valero. R, 2010, [http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes\\_RodrigoValeroR\\_Integraci%C3%B3nMandoInalambrico\\_2010.pdf?sequence=1](http://www.repositoriocdpd.net:8080/bitstream/handle/123456789/651/Tes_RodrigoValeroR_Integraci%C3%B3nMandoInalambrico_2010.pdf?sequence=1))



**Figura 14-1:** Chip Bluetooth del mando Wii

**Fuente:** <https://www.sparkfun.com/tutorials/43>

### 1.2.4 *Componentes externos del mando Wii*

#### 1.2.4.1 *Nunchuk*

Este dispositivo es el primer accesorio de Nintendo para el mando de Wii. Se conecta al mando de Wii mediante un cable que es aproximadamente de 1 a 1,2 m (3,5 a 4 pies) de largo. Se trabaja en conjunto con el controlador principal en muchos juegos. Al igual que el mando de Wii, el Nunchuk también proporciona un acelerómetro de tres ejes (x, y, z) de detección de movimiento e inclinación.

#### 1.2.4.1.1 Registros e inicialización

Los controles de extensión se mapean en la dirección 0xa40000 y tienen en total 100 bytes de longitud. Los registros se pueden leer y escribir. Los datos están cifrados, el código de encriptación se configura al principio del registro 0xa40040. Si se escribe un valor nulo, se inicia el periférico y hace uso de un esquema de encriptación muy sencillo. Si queremos recuperar la información original hemos de usar la siguiente transformación:  $\text{decrypted\_byte} = (\text{encrypted\_byte} \text{ XOR } 0x17) + 0x17$ . El resultado debe truncarse a 8 bits.

#### 1.2.4.1.2 Registros e inicialización de Identificación

Los últimos 2 bytes del bloque de registros para el controlador de expansión, identifican al dispositivo concreto conectado al puerto. Si se hace una lectura de 2 bytes en el registro 0xa400fe tendremos estos bytes. Podemos encontrar los valores registrados en la tabla 4-1.

**Tabla 4-1:** Inicialización e Identificación

Encriptado	Desencriptado	Significado
0x0000	0X2E2E	Nada Conectado
0xFFFF	0XFFFF	Parcialmente Conectado
0xFEFE	0x0000	Nunchuk conectado
0xFDFD	0x0101	Controlador clásico conectado

**Fuente:** <http://bing.us.es/proyectos/abreproy/11823/fichero/Volumen+I%252F5.pdf>

#### 1.2.4.1.3 Información De Estado

El mando nos puede indicar el estado de los dispositivos periféricos, el nivel de batería, etc.

Para solicitar dicha información hemos de enviar cualquier información al report 0x15.

La información de estado se recibe a través del report 0x20: 20 00 00 FF 00 00 BB

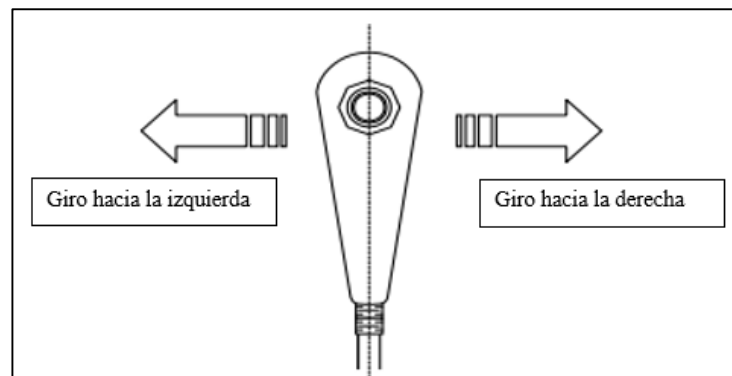
En ese contexto, 0xBB es el nivel de la batería y 0xFF es una máscara de banderas con el siguiente significado de estado:

**Tabla 5-1:** Información de estado

Bit	Mascara	Significado
0	0x01	No identificado
1	0x02	Periférico conectado
2	0x04	Altavoz
3	0x08	Envío Continuo solicitado
4	0x10	LED1
5	0x20	LED2
6	0x40	LED3
7	0x80	LED4

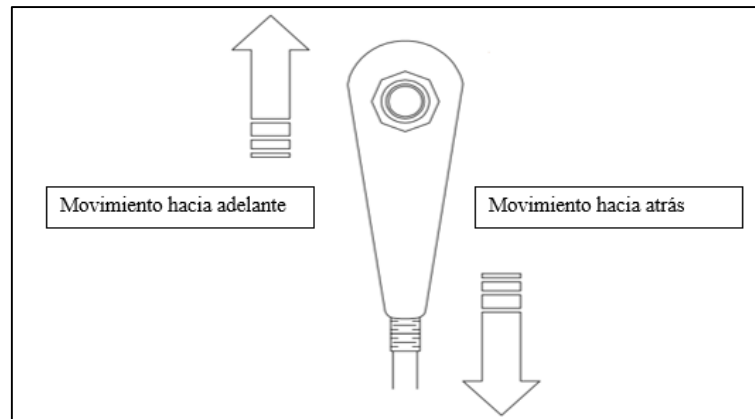
**Fuente:** <http://bibing.us.es/proyectos/abreproy/11823/fichero/Volumen+I%252F5.pdf>

#### 1.2.4.1.4 Movimientos del Nunchuk



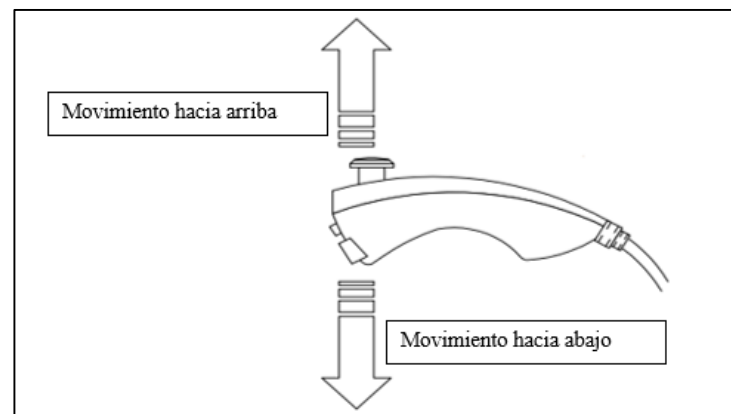
**Figura 15-1:** Movimiento hacia la derecha o izquierda.

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.



**Figura 16-1:** Movimiento hacia adelante o atrás

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



**Figura 17-1:** Movimiento hacia arriba o abajo

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 1.2.4.1.5 Partes del Nunchuk



**Figura 18-1** Partes del Nunchuk.  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

### **1.3 Lenguajes de Programación**

Es un lenguaje formal el cual permite realizar procesos que pueden ser llevados a cabo por máquinas como un ordenador, existen diversos lenguajes como por ejemplo; LABVIEW, Visual Studio C#, etc.

#### **1.3.1 LabView**

LabView nació en el año 1976 quien le dio vida fue la empresa National Instruments (NI) con el propósito de funcionar sobre máquinas Apple Macintosh (MAC). En la actualidad está disponible en los principales sistemas operativos utilizados en el mundo.

LabView (Laboratory Virtual Instrument Engineering Workbench) es un entorno gráfico de desarrollo de programación. Utiliza símbolos gráficos en lugar de lenguaje textual. Está totalmente integrado para la comunicación con hardware GPIB, VXI, etc. También cuenta con librerías para estándares de software como TCP/IP y ActiveX. (NI).

Este entorno de programación es muy dinámico y sencillo en relación a otros entornos de programación ya que el usuario puede crear aplicaciones sin línea de código, una característica es que se puede conectar con todo tipo de hardware incluyendo instrumentos de escritorio, tarjetas insertables, controladores de movimiento y controladores lógicos programables (PLCs). (Rodrigo Gonzales Cayufilo, Rodrigo Gonzales Pino, 2007, p. 15)

##### **1.3.1.1 Características**

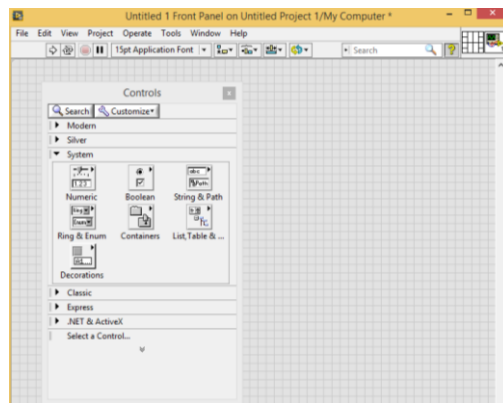
Hoy en día, científicos, ingenieros, técnicos y estudiantes utilizan LabVIEW para desarrollar soluciones que respondan a sus interrogantes, es por ello que su principal característica es la facilidad de uso que posee. También resulta válido para personas con pocos conocimientos en programación, ya que pueden realizar programas relativamente complejos, imposibles para ellos con los lenguajes tradicionales. (Rodrigo Gonzales Cayufilo, Rodrigo Gonzales Pino, 2007, p. 16)

LabView posee facilidad de manejo para las siguientes interfaces de comunicación: Puerto Serial y paralelo, GPIB, USB, PXI, VXI, TCP/IP, UDP, DataSocket, Irda, Bluetooth, OPC, etc.

### 1.3.1.2 Aspectos generales

#### 1.3.1.2.1 Panel frontal

Hace referencia a la interacción del usuario con el VI, Los controles son igual a las entradas y los indicadores a las salidas, también está formado de pulsadores, botones, etc.

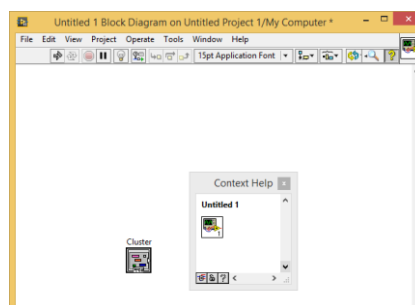


**Figura 19-1:** Panel Frontal.

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 1.3.1.2.2 Diagrama de Bloque

Se refiere al código que controla el programa, es decir el código fuente del VI. Es donde se controla el procesamiento de entradas y salidas que se crean en el panel frontal.



**Figura 20-1:** Diagrama de bloque

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Para construir un diagrama de bloques se debe conectar los distintos objetos entre sí, utilizando cables y uniendo los terminales de entrada y salida, el color varía dependiendo del tipo de dato que transporta. LabView cuenta con una gran biblioteca de funciones como: conversiones, comparaciones, funciones de entrada y salida, etc. Las estructuras son similares a los lenguajes de programación convencionales porque también utiliza los bucles (for, while, etc.) al momento de ejecutar el código.

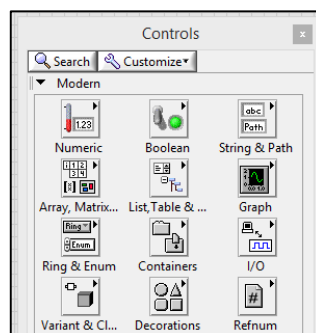
#### *1.3.1.2.3      Compilación del programa*

Al momento de programar en Labview, el lenguaje escrito de forma gráfica no es ejecutable por el ordenador, es por ello que primero se analiza el código fuente y se crea un código assembler, al momento de compilar Labview muestra errores en caso de haberlo, para saber dónde se da el problema basta con darle doble clic en el error y muestra el diagrama de bloques para que el programador pueda corregirlo.

#### *1.3.1.2.4      Paletas*

##### *1.3.1.2.4.1      Paletas de control*

La paleta de control podemos encontrar en el panel frontal, para construir el diagrama de bloques se usan elementos e indicadores los cuales podemos encontrar en el panel de control,



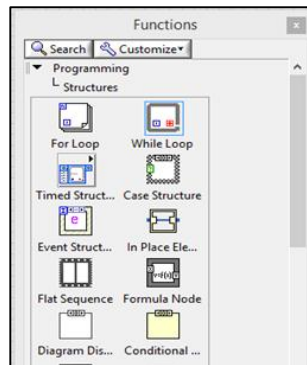
**Figura 21-1:** Paleta de control

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

##### *1.3.1.2.4.2      Paleta de funciones*



Para crear las funciones y estructuras encontramos en la paleta de funciones y VI's todos en conjunto se denominan nodos, para interconectar estos nodos se utiliza la herramienta de conexión (Wiring Tool) la cual se ubica en la paleta de herramientas.



**Figura 22-1: Paleta de funciones**  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 1.3.1.2.4.3 *Paleta de Herramientas*

La paleta está disponible en panel de control y en el diagrama de bloques ya que ayuda a modificar los contenidos, cuenta con herramientas de etiquetado, cableado, de operaciones, etc.

Si la herramienta de selección automática está habilitada y usted mueve el cursor sobre un objeto en el panel frontal o en el diagrama de bloque, LabVIEW automáticamente selecciona la herramienta correspondiente de la paleta de controles. (Tutorial de Labview, <http://www.esi2.us.es/~asun/LCPC06/TutorialLabview.pdf>)



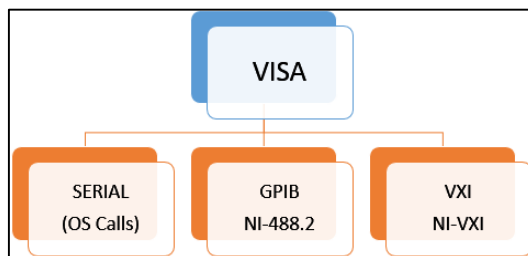
**Figura 23-1:** Paleta de herramientas  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 1.3.1.2.5 Comunicación serial

#### 1.3.1.2.5.1 VISA

La necesidad de interactuar entre el usuario y los distribuidores de sistemas de bus ha logrado crear esta herramienta que facilita la comunicación entre sí, en núcleo de estos programas son los VI (Instrumento Virtual). VISA permite al controlador vincular/interactuar con el dispositivo asociado, este software ayuda a visualizar información de los dispositivos GPIB y VXI conectados al ordenador. Labview clasifica los puertos de comunicación ASRL. Donde ASRL1 tiene conexión directa con el COM1, ASRL2 con el COM2, etc.

VISA por sí mismo puede programar instrumentos porque es un API de alto nivel las cuales son llamadas dentro de los drivers de bajo nivel, no toma en cuenta el tipo de interfaz al momento de comunicarse con instrumentos, utiliza las mismas operaciones en la siguiente figura podemos apreciar la jerarquía de VISA:

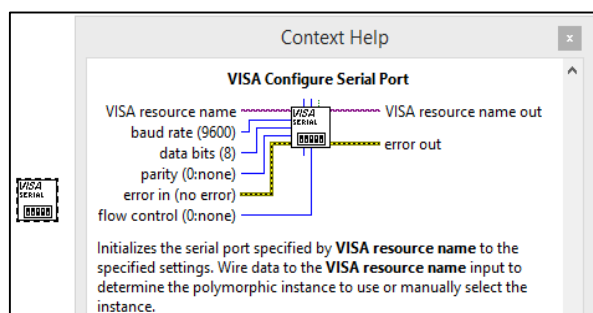


**Figura 24-1:** Puerto serial VISA

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 1.3.1.2.6 Visa Configure Serial Port

Permite la comunicación con el puerto COM del ordenador, ayuda a configurar la velocidad de transmisión, el número de baudios, etc.



**Figura 25-1:** VISA configure Serial Port

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

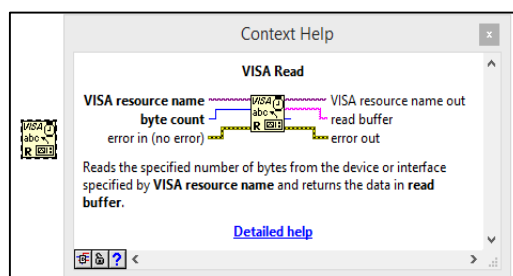
Cada ítem tiene su respectiva función, entre los más importantes:

- VISA resource name: es el que nos permite seleccionar el puerto lógico (COM).
- Baud rate: son los baudios que va a transmitir es decir la tasa de transferencia.

- Data Bits: es la cantidad de bits de entrada por defecto 8.
- Parity: es aquel que hace una comparación para saber si el bit de paridad es par o impar.
- Flow control: determina el control guiándose en la de transferencia de datos.
- Visa resource name out: es el retorno de la función.

#### 1.3.1.2.6.1 VISA lectura

Es el encargado de leer un número específico de bytes desde la interfaz especificada y como resultado arroja datos en el buffer de paridad.

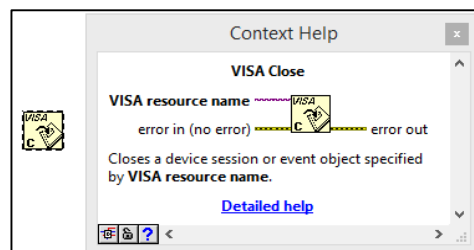


**Figura 26-1** VISA de lectura.

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 1.3.1.2.6.2 VISA cierre

Es el encargado de terminar la sesión del dispositivo conectado a la interfaz de entrada.

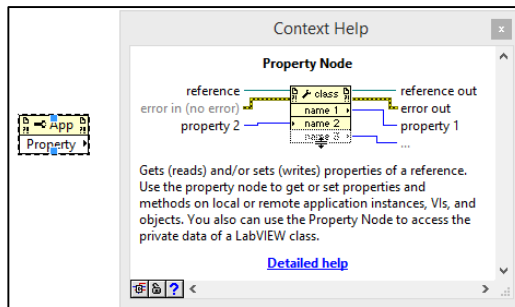


**Figura 27-1:** VISA de cierre

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 1.3.1.2.6.3 Nodo de propiedad

Este nodo realiza varias funciones lee/escribe las propiedades de una referencia, se utiliza para obtener o establecer las propiedades y métodos de instalaciones de aplicaciones locales o remotas, VI's y objetos. También se puede utilizar para tratar a los datos privados de una clase.



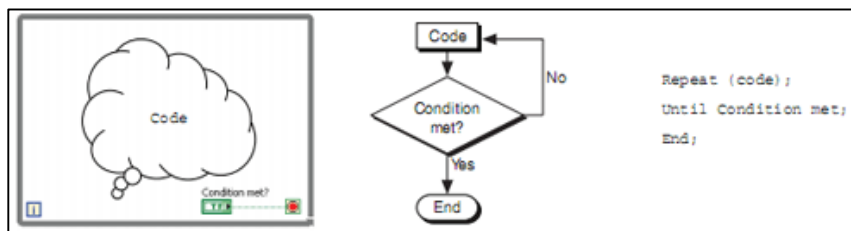
**Figura 28-1:** Detalles del nodo de propiedad.  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

### 1.3.1.3 Estructuras de LabVIEW

Las estructuras de Labview contienen código gráfico las cuales todo lo que contengan dentro de ella, cumplirá un ciclo repitiendo las veces que el programador cree necesario, es decir, hasta que se cumpla una condición.

#### ➤ Ciclos While

Similar a un Ciclo Do o a un Ciclo Repeat-Until en lenguajes de programación basados en texto, un Ciclo While, como ejemplo tomamos la figura 29-1, la cual ejecuta el código que contiene hasta que ocurre una condición. El Ciclo While está ubicado en la paleta Structures. (National Instruments, 2016, <http://www.ni.com/getting-started/labview-basics/esa/execution-structures>).



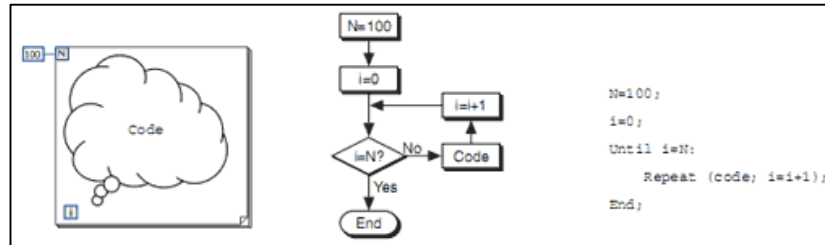
**Figura 29-1:** Ciclo While

Fuente: <http://www.ni.com/getting-started/labview-basics/esa/execution-structures>

En el primer bloque podemos observar el Ciclo While de LabVIEW, en el segundo el Diagrama de Flujo y el tercer bloque tenemos el Pseudo Código.

#### ➤ Ciclo For

Un Ciclo For ejecuta un subdiagrama un número de veces establecido. Como podemos apreciar en la figura muestra un Ciclo For, un diagrama de flujo equivalente a la funcionalidad y un pseudo código del Ciclo For. El Ciclo For está en la paleta Structures.



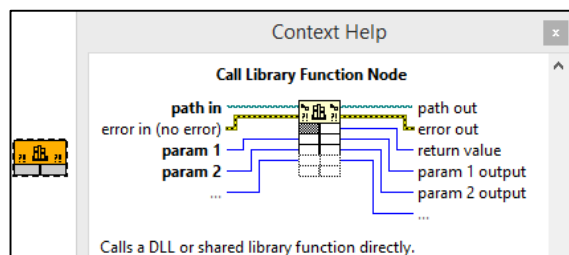
**Figura 30-1:** Ciclo For

**Fuente:** <http://www.ni.com/getting-started/labview-basics/esa/execution-structures>

También puede colocar un Ciclo While en el diagrama de bloques, dar clic derecho en el Ciclo While y seleccionar Replace with For Loop en el menú de acceso directo para cambiar un Ciclo While Loop a Ciclo For. (National Instruments, 2016, <http://www.ni.com/getting-started/labview-basics/esa/execution-structures>).

#### 1.3.1.4 Call Library Function Node

Labview tiene una funcionalidad la cual permite utilizar los dll con mucha facilidad, esta librería se conoce con el nombre de Call Library Function Node es compatible con un gran número de tipos de datos y convenciones de llamada. Si desea llamar a una DLL que contiene objetos ActiveX, se utiliza la función Open Automation con el Property Node y el Invoke Node. (National Instruments, 2016, [http://zone.ni.com/reference/en-XX/help/371361H-01/glang/call\\_library\\_function/](http://zone.ni.com/reference/en-XX/help/371361H-01/glang/call_library_function/)).



**Figura 31-1:** Call Library Function Node

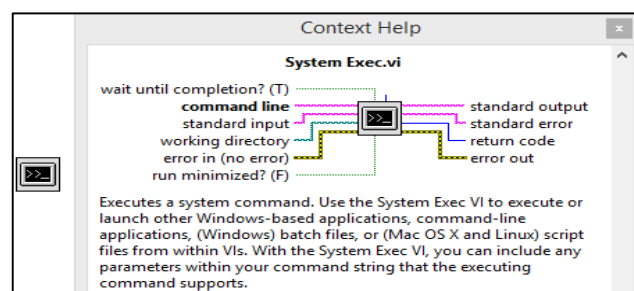
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

Cada ítem tiene su respectiva función:

- Path in: Identifica el nombre o la ruta de la biblioteca compartida que desea llamar.

- Error in: Describe las condiciones de error que se producen antes de ejecutar esta librería.
- Param 1..n: Son los ejemplos de los parámetros de entrada de la función de la librería.
- Path out: devuelve la ruta de la llamada DLL o la librería compartida
- Error auto: contiene información de error de la función de la librería.
- Return value: es un ejemplo del valor de retorno
- Param 1...n output: son ejemplos de los parámetros de salida en función a la librería.

#### 1.3.1.5 *System exec.vi*



**Figura 32-1: System Exec.vi**

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

Donde los ítems son:

- Use expected output size: Mejora la eficiencia de memoria
- Command line: Indica el comando de Labview que llama a ejecutar un programa.
- Standard input: Es el texto para pasar a la línea de comandos como entrada estándar.
- Error in: Describe las condiciones de error que se producen antes de ejecutar este nodo.
- Standard output: Devuelve información del programa que llama.
- Standard error: devuelve errores del programa que llama.
- Return code: Indica el código de salida system-dependent.
- Error out: Contiene información de error.

#### 1.3.2 *Visual Studio C#*

Visual Studio C# es un lenguaje orientado a objetos que está diseñado para el desarrollo de una variedad de aplicaciones que se ejecutan en la plataforma .NET, la cual es propia de Microsoft, este lenguaje de programación toma las mejores características de lenguajes de versiones anteriores y los combina en uno solo, Microsoft ha escrito la mayor parte de la BCL usándolo,

por lo que su compilador es el más depurado y optimizado de los incluidos en el .NET (González. J, <http://dis.um.es/~bmoros/privado/bibliografia/LibroCsharp.pdf>)

#### 1.3.2.1 *Características*

➤ Sencillez:

El código escrito en C# es auto contenido, lo que significa que no necesita de ficheros adicionales al propio fuente tales como ficheros de cabecera. El tamaño de los tipos de datos básicos es fijo e independiente del compilador, sistema operativo o máquina para quienes se compile, lo que facilita la portabilidad del código. C# no cuenta con elementos innecesarios en .NET, además incluye elementos que han de mostrado ser valiosos en otros lenguajes de programación. No se incluyen elementos tales como macros, herencia múltiple, etc.

➤ Modernidad:

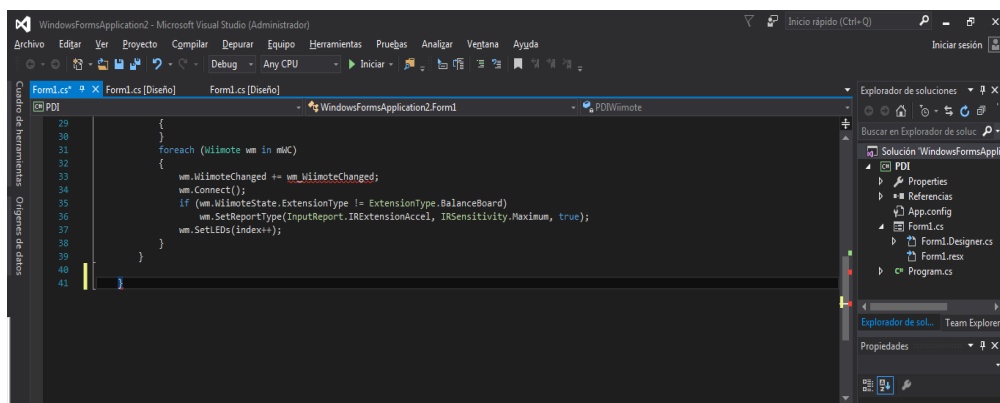
C# incorpora en el propio lenguaje elementos que a lo largo de los años ha ido demostrándose son muy útiles para el desarrollo de aplicaciones y que en otros lenguajes como Java o C++ hay que simular, la inclusión de una instrucción “for each” que permita recorrer colecciones con facilidad y es ampliable a tipos definidos por el usuario, la inclusión de un tipo básico “string” para representar cadenas o la distinción de un tipo “bool” específico para representar valores lógicos. (Medina. E. 2005, <http://repositorio.espe.edu.ec/bitstream/21000/631/1/T-ESPE-012138.pdf>)

➤ Orientación a objetos

Como todo lenguaje de programación de propósito general actual, C# es un lenguaje orientado a objetos, aunque eso es más bien una característica del CTS que de C#. Una diferencia de este enfoque orientado a objetos respecto al de otros lenguajes como C++ es que el de C# es más puro en tanto que no admiten ni funciones ni variables globales sino que todo el código y datos han de definirse dentro de definiciones de tipos de datos. (Medina. E. 2005, <http://repositorio.espe.edu.ec/bitstream/21000/631/1/T-ESPE-012138.pdf>)

#### 1.3.2.2 *Funcionalidades.*

- Orientación a componentes: la sintaxis en C# permite definir cómodamente propiedades, eventos o atributos.
- Gestión automática de memoria: todo lenguaje de .NET tiene cuenta con el recolector de basura.
- Incluye delegados, que son similares a los punteros a funciones de C++ pero siguen un enfoque orientado a objetos, pueden almacenar referencias a varios métodos simultáneamente.
- Extensibilidad: de tipos básicos, de operadores y de modificadores.
- Eficiente: En la plataforma todo el código incluye numerosas restricciones para asegurar su seguridad y no permite el uso de punteros.
- Compatible: para facilitar la migración de programadores, el CLR (Common Language Runtime) también ofrece, a través de los llamados Platform Invocation Services (PInvoke), la posibilidad de acceder a código nativo escrito como funciones sueltas no orientadas a objetos tales como las DLLs de la API Win32.



**Figura 33-1:** Plataforma del Visual Studio C#

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

### 1.3.2.3 Elementos Importantes

#### ➤ Clases y Objetos

Un objeto es un agregado de datos y de métodos que permiten manipular dichos datos, y un programa en C# es un conjunto de objetos que interaccionan unos con otros a través de sus métodos. Una clase es la definición de las características concretas de un determinado tipo de objetos.

#### ➤ Espacios De Nombres

Del mismo modo que los ficheros se organizan en directorios, los tipos de datos se organizan en espacio de nombres. Esto permite tener organizados los tipos de datos, facilitando su localización.



➤ Eventos y delegados

Permiten controlar la forma en que se accede a los campos delegados, en general podemos decir que los eventos proporcionan un medio para que una clase u objeto informe a otras clases u objetos cuando sucede algo relevante. Un delegado es una clase especial cuyos objetos pueden almacenar referencias a uno o más métodos de esta forma a través del objeto sea posible solicitar la ejecución en cadena de todos ellos.

➤ Programación Multi-hilo

Se conoce como flujo de control secuencial dentro de un programa o hilo. Al igual que un proceso, puede gozar de cierta autonomía pero a diferencia de un proceso, diversos hilos de una aplicación pueden compartir los mismos datos. Estos hilos no se pueden correr dentro de un programa secuencial, para ello se ejecuta dentro de un programa. Se pueden programar múltiples hilos de ejecución para que corran simultáneamente en el mismo programa. (LA WIIMOTELIB, <http://bibing.us.es/proyectos/abreproy/11823/fichero/Volumen+I%252F4.pdf>).

#### 1.3.2.4 Librería WiimoteLib

Es una librería de Microsoft .NET cuyo objetivo es el de reconocer el mando Wii Remote, el inicio de este proyecto fue en el año 2007 en una página web de Microsoft, como resultado se obtuvo una interfaz de programación de aplicaciones (API), fácil usar que puede ser utilizada en cualquier aplicación gestionada. La librería diseñada para C# contiene archivos con extensión.cs, estos archivos definen las estructuras y funciones que ayudan a controlar el Wiimote. A continuación se describe cada uno de los archivos y su respectiva función:

##### 1.3.2.4.1 *DataTypes.cs*

Aquí nos ayuda a identificar el estado del mando Wii, esto se logra gracias a que cuenta con estructuras simples que finalmente crean una clase.

➤ Clase estado del Wiimote

La interacción con el mando es uno de los puntos más importantes es por ello que esta clase cuenta con estructuras que recolectan toda clase de información relacionada con el mando Wii.

➤ Estructuras

Las estructuras que conforman la clase del estado del Wii remote se definen a continuación:

➤ Estado de los botones (ButtonState)

Aquí contiene el estado actual de los botones del Wiimote, muestra si el botón está siendo utilizado o no, cada botón cuenta es representado por una variable de tipo bool. Los botones son: A, B, Plus, Home, Minus, One, Two, Up, Down, Left y Right.

➤ Estado de los LED's (LEDState)

Cada led se representa mediante una variable bool que indica si está encendido o apagado. Las variables son: LED1, LED2, LED3 y LED4.

➤ Estado de la cámara infrarroja (IRState)

Almacena el estado actual de la cámara infrarroja.

#### 1.3.2.4.2 *Events.cs*

Inicialmente la librería se pensaba trabajar con eventos de ahí su nombre Events, este archivo define dos clases las cuales serán heredadas de EventArgs, esta es la clase base para clases que contengan datos de algún evento. A continuación se detalla cada una de las clases:

➤ Clase WiimoteExtensionChangedEventArgs

Esta clase es el argumento pasado al evento WiimoteExtensionChangedEvent. Dicho evento se activa cuando detecta la conexión/desconexión en el puerto periférico, un ejemplo puede ser cuando se conecta/desconecta el nunchuck.

➤ Clase WiimoteChangedEventArgs

Esta clase es el argumento pasado al evento WiimoteChagedEvent, este se activa cuando el estado del Wiimote cambia. El estado actual del Wiimote es representada por el único objeto que posee esta clase, WiimoteState.

#### 1.3.2.4.3 *HIDImports.cs*

El intercambio de datos en este archivo se hace a través de reportes. Las estructuras tienen un cierto formato flexible que pueden manejar muchos tipos de datos. El tamaño de cada report es fijo. Para responder a los datos de un report, se debe saber que report usar para responder, existen tres tipos de reports: de entrada, salida y de características. HID es una de las clases que nos

aporta USB, en este existe distintos tipos de transferencias. Como es el caso de HID el cual se usan las transferencias de control y por interrupción.

#### *1.3.2.4.4 Wiimote.cs*

El archivo más importante considerado como principal es Wiimote.cs. Aquí se define la clase Wiimote, en la cual se incluyen la definición de los eventos, constantes, variables y métodos para el manejo del Wiimote.

##### ➤ Eventos

WiimoteChanged como su nombre lo indica, es el evento que se activa cuando existe algún cambio en el estado del MandoWii, para realizar cualquier aplicación es importante que el dispositivo realice barridos periódicamente, es decir actualiza constantemente los parámetros, es por ello que a este evento se considera como principal.

#### *1.3.2.4.5 WiimoteCollection.cs*

Este archivo es multifuncional porque permite trabajar con varios Wiimotes a la vez, esto se logra gracias a la clase WiimoteCollection la cual utiliza dos métodos que son: FindAllWiimotes; es el encargado de detectar los Wiimotes y WiimoteFound; la función es añadir los Wiimotes detectados al sistema. (LA WIIMOTELIB, <http://bibing.us.es/proyectos/abreproy/11823/fichero/Volumen+I%252F4.pdf>)

## **CAPITULO II**

## **2 MARCO METODOLÓGICO**

## **2.1 Introducción**

En este capítulo se describe el marco metodológico que se desarrolló en el trabajo de titulación, mostrando aspectos principales como la selección del hardware, software, etc. Además se presenta el diseño e implementación de una PDI con control inalámbrico, con el fin de concluir los objetivos propuestos.

En la parte de diseño constará de una explicación del funcionamiento total del dispositivo, una vez escogida la alternativa ideal que mejor se ajuste a nuestros requerimientos, se procederá a describir cada uno de los componentes de la parte Hardware, software y sus respectivas aplicaciones. La implementación es la parte más detallada donde se dará a conocer paso a paso las configuraciones, conexiones y la programación de los componentes y softwares.

### **2.1.1 Métodos de investigación**

Los métodos que se tomaron en cuenta para la selección del hardware y software en el desarrollo del presente proyecto están basados en ciertos factores importantes que se deben considerar a la hora de la elección, tanto físicos como lógicos que pueden llegar a ser indispensables para su correcto funcionamiento, con el fin de cumplir las necesidades requeridas con las mejores prestaciones en cuanto a relación beneficio/costo, los métodos utilizados son:

#### **➤ Método comparativo**

El método comparativo nos ayuda analizar de manera sistemática dos o más dispositivos basados en tecnología infrarroja, para establecer sus similitudes y diferencias sacar conclusiones tomando en cuenta las variables que poseen cada uno.

#### **➤ Método inductivo**

El método inductivo es aquel que obtiene conclusiones generales a partir de premisas particulares, la clasificación y el estudio nos permitirá llegar a una generalización y a una correcta elección del software.

## 2.2 Selección de Hardware

### 2.2.1 *Análisis y elección del hardware*

Para el desarrollo del proyecto se consideró principalmente el tipo de tecnología que presenta el dispositivo que se pretende utilizar ya que nuestro trabajo se basa en el uso de la tecnología infrarroja para su desarrollo.

La tecnología infrarroja se basa en rayos luminosos los cuales se desplazan a lo largo del espectro electromagnético, el estándar IrDA permite la comunicación entre dos terminales a velocidades que oscilan entre los 9600 bps y los 4Mbps, cabe recalcar que soporta diversos dispositivos de comunicaciones, electrónicos, etc.

Entre las principales características que ofrece esta tecnología tenemos un alcance que cubre distancias significativas, conexión inalámbrica, comunicación punto a punto y soporta varias plataformas de hardware y software. Por lo cual buscamos dispositivos que cuentan con dicha tecnología previo a su selección.

#### 2.2.1.1 *Selección del dispositivo infrarrojo*

En el mercado existen varias opciones de hardware que están fabricados en base a tecnología infrarroja, para ello debemos realizar un análisis previo para seleccionar el dispositivo de mejores prestaciones, fácil adquisición y de bajo costo, entre ellos tenemos los siguientes:

- Mando Wii (Wii Remote).
- Kinect.
- Leap Motion.

##### 2.2.1.1.1 *Ponderación*

La Ponderación consiste en asignar pesos a cada variable, iniciando desde inadecuado/malo un con peso de cero hasta muy adecuado que es la mejor opción y se asignó un peso de tres.

**Tabla 1-2:** Ponderación para elección de hardware

ESCALA CUALITATIVA			
Muy adecuado	Adecuado	Poco adecuado	Inadecuado
(3)	(2)	(1)	(0)

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 2.2.1.2 Parámetros concernientes

##### 2.2.1.2.1 Precio

Esta variable se determina considerando el componente cuyo valor sea el más económico y a su vez cuente con los requerimientos necesarios para obtener resultados similares en su implementación, de esta forma obtenemos el mismo resultado pero económicamente más barato.

Si el precio es menor a \$50 tendrá un peso igual a 3 que representa el dispositivo más económico, mientras que si el valor del dispositivo está entre \$50 y \$100 equivale a un peso de 2 y si el valor es mayor a \$150 será 1.

**Tabla 2-2:** Precios de los dispositivos

DISPOSITIVO	PRECIO	CUANTIFICACIÓN
Kinect	\$160	1
Mando Wii	\$40	3
Leap Motion	\$75	2

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

##### 2.2.1.2.2 Características

Esta variable se fijará de acuerdo a los requerimientos necesarios que deben estar presentes en el dispositivo para el desarrollo de la Pizarra digital interactiva, entre los principales se consideran: sensor de emisiones infrarrojas, acelerómetro, emisor y receptor Bluetooth.

Si el dispositivo cuenta con esa característica le corresponde un valor de 1, con lo cual al disponer con todas las características necesarias obtendría un valor sumado de 3, en cambio se puntúa con un 0 cuando dicha característica no está presente en el dispositivo.

**Tabla 3-2:** Características de los dispositivos

DISPOSITIVO	CARACTERISTICAS			
	<i>E/R Bluetooth</i>	<i>Acelerómetro</i>	<i>Sensor Infrarrojo</i>	<b>Total</b>
Kinect	0	0	1	1
Mando Wii	1	1	1	3
Leap Motion	1	1	0	2

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 2.2.1.2.3 Actualización y mejoras del dispositivo

Esta variable se determina mediante el continuo desarrollo e innovación que posean los fabricantes del dispositivo, es decir aquellos que se encuentren actualmente operando en el mercado ofreciendo soportes y actualizaciones periódicamente, además la disposición de disposición de software que permita su utilización.

De la misma forma, si el dispositivo cuenta con esta característica se asigna un valor de 3 caso contrario 0, no hay valor intermedio debido a que es un factor indispensable para su manipulación y la factibilidad que presenta su utilización.

**Tabla 4-2:** Actualización de los dispositivos

DISPOSITIVOS	ACTUALIZACION
Kinect	3
Mando Wii	3
Leap Motion	0

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Teniendo en cuenta las diferentes características, precios y ponderaciones descritas anteriormente obtenemos el siguiente resultado.

#### 2.2.1.3 Resultados en la elección del Hardware

Para la decisión del dispositivo que vamos a utilizar, nos basaremos en la interpretación de los datos expuestos en la siguiente tabla.

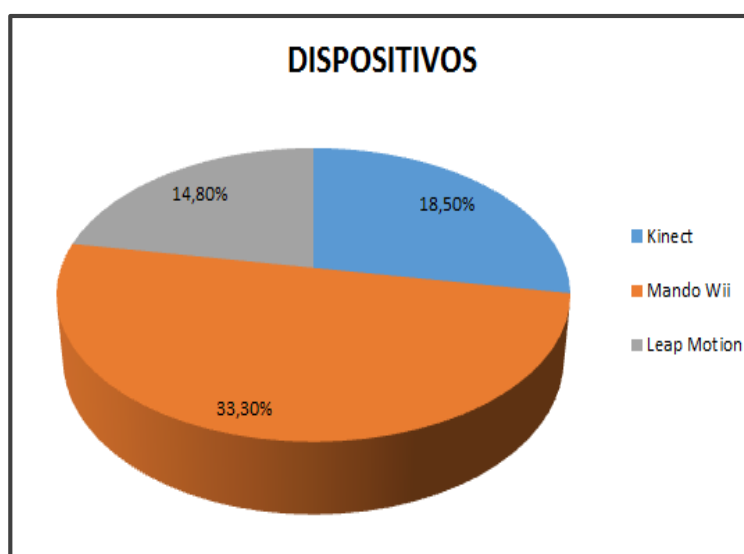
**Tabla 5-2:** Análisis para la selección de hardware

DISPOSITIVOS	PARAMETROS						
	Precio	Características	Actualización	SUMA		%	
Kinect	1	1	3	5	9	18,50%	33,33%
Mando Wii	3	3	3	<u>9</u>	9	<u>33,30%</u>	33,33%
Leap Motion	2	2	0	4	9	14,80%	33,33%
<b>Total</b>				18	27	66,60%	100%

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Para realizar la elección del dispositivo a usar, se tomó en cuenta el porcentaje más alto ya que es el mejor y cumple con las características necesarias dando como ganador al Mando Wii.

El valor numérico perfecto es aquel resultado de la sumatoria de los parámetros tomando en cuenta las equivalencias correspondientes, para una mejor apreciación de los valores tenemos la siguiente gráfica:



**Figura 1-2** Resultado para elección de dispositivos

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

### 2.2.2 Etapa Hardware



### 2.2.2.1 *Wii Motion Plus*

Es un accesorio de la consola de video juegos de Nintendo conocido como Wii Motion y actualmente contamos con la nueva versión Plus, la cual tiene una mejora que aumenta la precisión respecto de la versión anterior captando movimientos más complejos en tiempo real, todo esto se logra gracias al sensor que viene incorporado en el dispositivo y la barra de sensores del mismo, además cuenta con las siguientes características:

- ✓ Detecta velocidades de giro en los tres ejes: vertical, longitudinal y lateral.
- ✓ Reconoce giros y cambios de ángulo de forma inmediata.
- ✓ Sensor giroscópico, también es usado por aviones o barcos como estabilizador.

### 2.2.2.2 *Nunchuk*

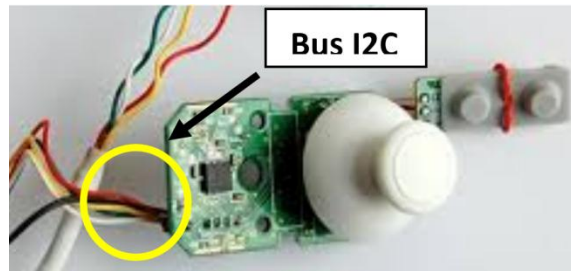
El nunchuck presenta un menor número de componentes y controles en comparación al mando Wii, sin embargo se escogió utilizar este componente externo debido a que nos proporciona todos los movimientos que necesitamos para el desarrollo del control inalámbrico mediante el joystick que controla los desplazamientos en el eje x y en el eje y, además de los dos botones C y Z que son necesarios para la aplicación.

Este componente utiliza un bus de comunicación serial denominado bus I2C 1, el cual presenta una línea de datos (SDA) y una línea de reloj (SCL) ambas son bidireccionales y las más fundamentales, la comunicación con el nunchuk se puede lograr mediante la programación en arduino, la conexión entre el nunchuk y el arduino se mostrara más adelante.

**Tabla 6-2:** Función de los hilos del cableado

Color del hilo	Función
Rojo	Alimentación de 3,3 - 5 V
Blanco	Tierra (Ground)
Verde	Flujo de datos (SDA)
Amarillo	Clock (SCL)

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.



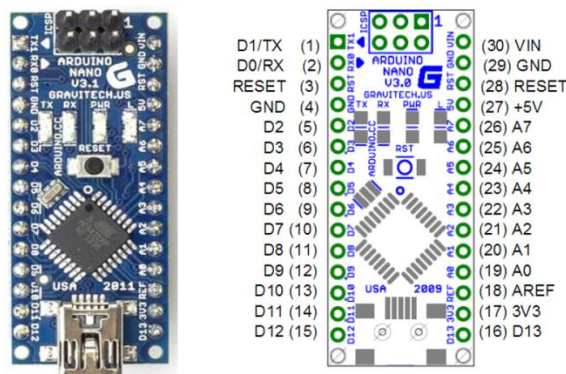
**Figura 2-2:** Bus I2C del Nunchuk  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



**Figura 3-2:** Partes internas de Nunchuk  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

### 2.2.2.3 *Arduino Nano*

Este dispositivo tiene las mismas funcionalidades que otros modelos la diferencia es como su nombre lo indica el más pequeño de su clase, posee pocos pines lo cual es una limitante para realizar proyectos grandes pero en nuestro caso es ideal ya que únicamente utilizaremos 2 pines. (ARDUINO.CC, [tps://www.arduino.cc/en/Main/ArduinoBoardNano](https://www.arduino.cc/en/Main/ArduinoBoardNano)).

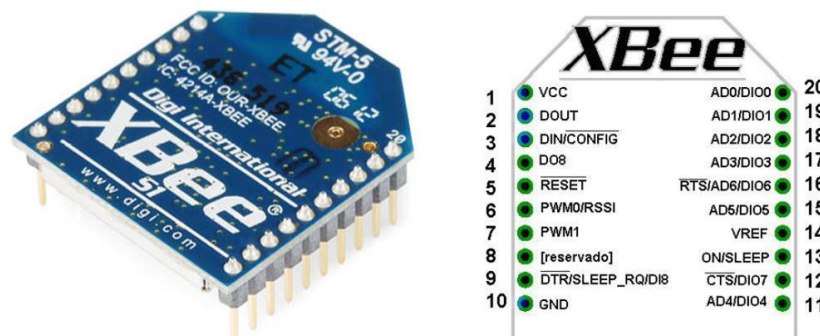


**Figura 4-2:** Arduino Nano y diagrama de pines  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Los pines a utilizar en nuestro proyecto son en #4 que cuya función es de la de conexión a tierra (Gnd), el #5 que es de in/out el cual nos servirá para el ingreso de información proveniente del nunchuk.

#### 2.2.2.4 Módulos Xbee

Los módulos Xbee cuentan con el protocolo Zigbee el cual realiza comunicaciones inalámbricas basado en el estándar IEEE802.15.4, este protocolo fue creado por una importante organización la cual no tiene fin de lucro (Zigbee Alliance), la transmisión de datos es a baja velocidad siendo la tasa de transferencia máxima y mínima de 20Kbps y 250Kbps respectivamente, el alcance depende del modelo de Xbee, la banda en la cual opera es la de 2,4Ghz siendo la que más se utiliza ya que está en la banda libre, pero también puede operar a 868 MHz y 915 MHz.



**Figura 5-2:** Módulo Xbee y diagrama de pines

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Cada módulo Xbee posee una dirección única y de igual forma existen dos series los cuales pueden ser configurados en los modos AT y API, los módulos de la serie 1 son utilizados para la configuración de redes punto a punto y punto a multipunto, y los módulos de la serie 2 que se utilizan para aplicaciones con repetidoras y redes mesh.

#### 2.2.2.5 Adaptador Bluetooth SCR

Bluetooth es la norma que define un estándar global de comunicación inalámbrica, que posibilita la transmisión entre diferentes equipos mediante un enlace por radiofrecuencia.

Los principales objetivos son:

- Facilitar las comunicaciones entre equipos móviles y fijos.
- Eliminar cables y conectores entre equipos.
- Crear pequeñas redes inalámbricas.

Como ejemplo tomamos el adaptador Bluetooth USB 4.0, es tan pequeño que puede dejarlo conectado al puerto USB del portátil sin que suponga un problema para transportarlo donde quiera. Con el adaptador USB 4.0 se conectará fácilmente y sin cables a cualquier modelo de auriculares Bluetooth, o para controlar sus presentaciones de PowerPoint, sin cables por medio del dispositivo adecuado, etc.

Este dispositivo es compatible con Windows 7, 8, 8.1, Vista, 10. Además con la mayoría de dispositivos Bluetooth como son (v1.1, v1.2, v2.0, v2.1 y los dispositivos v4.0 más avanzados) los requerimientos es tener disponible un puerto USB 2.0 o 1.2.



**Figura 6-2:** Adaptador de Bluetooth

**Fuente:**[http://redgear.co.uk/images/Public/Products/Bluetooth%20Dongles/White\\_CSRv4\\_0\\_dongle/White\\_CSRv4\\_0\\_dongle3.jpg](http://redgear.co.uk/images/Public/Products/Bluetooth%20Dongles/White_CSRv4_0_dongle/White_CSRv4_0_dongle3.jpg)

#### 2.2.2.6 *Puntero Infrarrojo*

Este elemento es un dispositivo que utilizaremos para manejar la pantalla proyectada por el infocus, su elaboración es sencilla y sus materiales son muy comunes y económicos. La forma y el tamaño no es de consideración sin embargo, su principal componente es un diodo infrarrojo de alta potencia.

## 2.3 Selección de Software

### 2.3.1 *Elección de Software y plataformas a usar*

El desarrollo de la aplicación para reconocer una PDI presenta varios requerimientos los cuales tomamos una decisión tomando en cuenta lo siguiente:

Lenguaje de Programación visual C#, es una característica a tomar en cuenta debido a la necesidad de obtener un código fuente capaz de interpretar y ejecutar los requerimientos propuestos anteriormente.

Lenguaje de programación de alto nivel LABVIEW, cuenta con una interfaz Gráfica, la cual facilita el trabajo al programador convirtiendo en una ambiente más amigable. Las librerías es un buen punto a tener en cuenta debido que al momento de programar la aplicación ayuda mucho ya que es un código fuente predefinido para reconocer ciertos tipos de hardware.

### **2.3.2 Etapa software**

#### **2.3.2.1 Visual Studio C# 2015**

Con la nueva experiencia de inicio de sesión simplificada diseñada para simplificar en gran el acceso a recursos en línea, incluso si tiene varias cuentas de Visual Studio. Al iniciar sesión en Visual Studio, automáticamente se inicia su sesión en todas las instancias de Visual Studio 2015 y Blend en su máquina. En Visual Studio 2015, su cuenta se comparte entre funciones para que, siempre que tenga un buen token, pueda obtener acceso a sus cuentas de Visual Studio Team Services desde Team Explorer y a recursos y sitios web desde su suscripción de Microsoft Azure.

- Control de versiones Git: es más fácil trabajar con bifurcaciones.
- Application Insights: detecta y diagnostica problemas de rendimiento y disponibilidad.
- Prueba: crea pruebas unitarias para el código Explore su código de .NET

#### **2.3.2.2 LabView 2015**

Esta versión de Labia continúa siendo compatible (estandarizado) es por ello que los usuarios pueden interactuar con casi cualquier hardware, mediante la reutilización del mismo código y los mismos procesos de ingeniería en los sistemas, ahorrando tiempo y dinero, además cuenta con las siguientes características:

- Es muy intuitivo y fácil de aprender.
- Los usuarios tiene la posibilidad de crear soluciones completas y complejas.
- Cuenta con un compilador gráfico para lograr la máxima velocidad de ejecución, etc. (2016).

### 2.3.2.3 *Arduino Nightly*

Es una plataforma de código abierto basado en hardware y software y su manipulación es fácil de usar. El hardware Arduino es capaz de interpretar datos enviados por sensores como; la luz, la huella dactilar mensajes del texto y convertirlo en datos de salida de acuerdo a nuestros requerimientos. (ARDUINO.CC, <https://www.arduino.cc/en/Guide/Introduction>).

A medida que el tiempo pasa, Arduino ha crecido cada vez innovándose y adaptando a las necesidades cotidianas de muchos proyectos realizados por estudiantes, programadores, profesionales e incluso por aficionados. Arduino tuvo gran impacto y gracias a su código abierto, la comunidad ha contribuido a su mejora. (ARDUINO.CC, <https://www.arduino.cc/en/Guide/Introduction>).

### 2.3.2.4 *Programa XCTU*

El X-CTU es una aplicación basada en Windows, este programa está diseñado para interactuar con los archivos de firmware que se encuentran en los productos de Digi RF y en los módulos Xbee y proporcionar una interfaz gráfica fácil y sencilla de usar.

En la ventana principal presenta las siguientes opciones:

- PC Settings: Permite seleccionar el puerto y la configuración del mismo
- Range Test: Permite desarrollar un test de rango entre dos dispositivos
- Terminal: Permite acceso a puerto COM del PC con un programa de emulación terminal.
- Modem Configuration: Permite programar los parámetros de los radios mediante una interfaz gráfica de usuario. Además permite cambiar la versión del firmware del radio

## 2.4 **Diseño del Algoritmo**

### 2.4.1 *Introducción*

El algoritmo permite al Mando Wii captar un haz de luz infrarroja y seguir su movimiento, convirtiendo en valores numéricos que indica la ubicación del punto en una superficie plana, los cuales son interpretados por el ordenador y posteriormente visualiza la localización del cursor

en la misma, dependiendo el caso el sistema ejecuta algún proceso o acción sobre el punto señalado por el puntero.

El algoritmo del control inalámbrico permite interpretar las acciones del nunchuck en instrucciones del ordenador de acuerdo a la acción que se realice, los datos se reciben mediante conexiones con dispositivos adicionales descritos anteriormente y se procesan en su aplicación respectiva, la cual en conjunto con la aplicación de la PDI se desarrollan posteriormente.

## **2.4.2 Desarrollo de la Pizarra Digital Interactiva**

### **2.4.2.1 Diseño del puntero Infrarrojo**

Dispositivo que se utiliza para interactuar con la imagen proyectada en la pizarra, cuenta con la función de recrear una instrucción de clic, en este caso el clic izquierdo, y además otras acciones que comúnmente realiza el puntero del ordenador como seleccionar, arrastrar y doble clic.



**Figura 7-2: Puntero Infrarrojo**

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

Esta constituido básicamente con un Diodo Infrarrojo, un par de pilas de 5V, resistencias de bajo valor nominal y una carcasa de marcador de pizarra, o cualquier contenedor acorde a su preferencia. A continuación sus materiales y precios:

**Tabla 7-2: Materiales y costo del Puntero Infrarrojo**

<b>Material</b>	<b>Costo Aproximado</b>
Led Infrarrojo	\$ 2,00
Pilas AA 1.5 V	\$ 1,50
Pulsador	\$ 0,50
Marcador (carcasa)	\$ 1,00
<b>Total:</b>	<b>\$ 5,00</b>

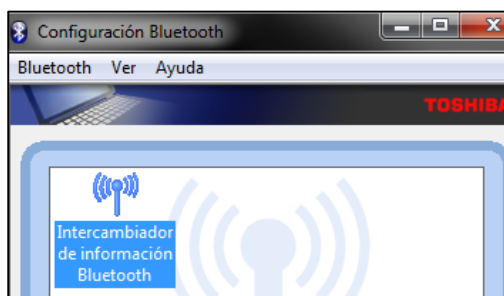
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

#### 2.4.2.2 Conexión del Mando Wii con el Ordenador.

Para realizar el intercambio de información es necesario establecer el medio de comunicación, para este proceso la comunicación se realiza de forma inalámbrica mediante Bluetooth para la transición y recepción de datos, y la ejecución de instrucciones.

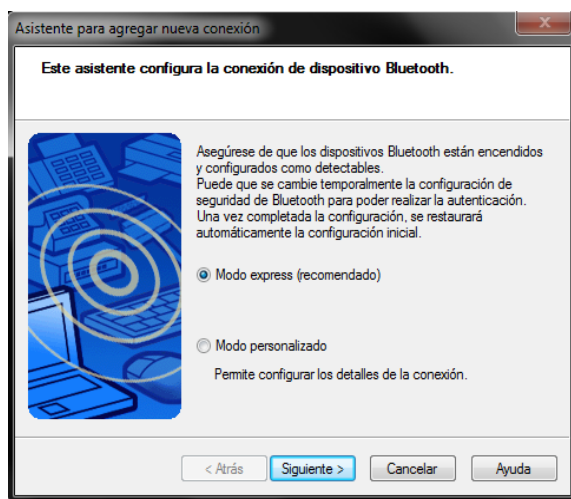
##### 2.4.2.2.1 Computador Portátil

- Se ingresa a la aplicación de Administración y Configuración del Bluetooth y se inicia una nueva conexión.



**Figura 8-2:** Búsqueda de dispositivos Bluetooth

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



**Figura 9-2:** Asistente de configuración Bluetooth.

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

- Para realizar la vinculación, se pulsa el botón de sincronización (SYNC) ubicado en la parte posterior del Mando Wii, el mismo se mostrará en la lista de dispositivos disponibles.





**Figura 10-2:** Botón SYN de Mando Wii.

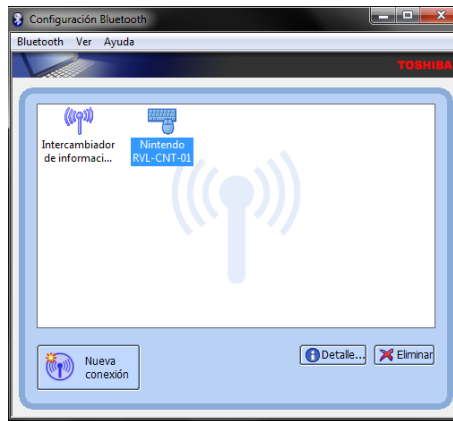
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



**Figura 11-2:** Selección de dispositivo Bluetooth

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Una vez que se realice la sincronización, el Mando Wii se visualizará en la Configuración de Bluetooth como un dispositivo listo para usarse, en algunos casos se solicita un código PIN para su vinculación, el mismo no es necesario y se prosigue normalmente.



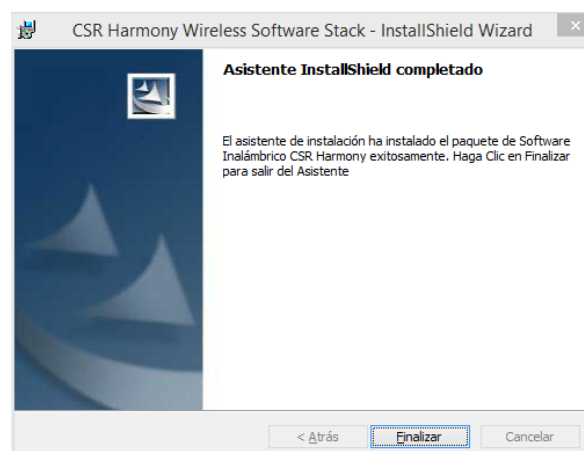
**Figura 12-2:** Dispositivo Bluetooth sincronizado  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

#### 2.4.2.2.2 Computadora de Escritorio (PC)

Para la sincronización con el Wii Mando es necesario la incorporación de un Bluetooth Externo, su instalación es breve y sencilla, sin ningún inconveniente.

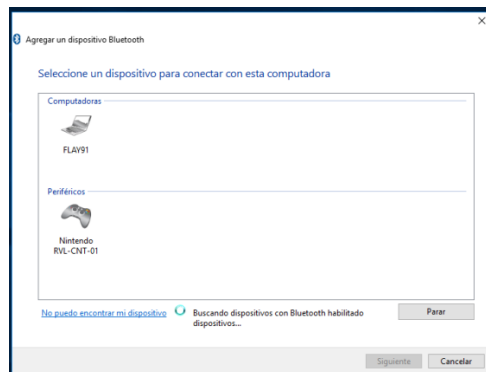


**Figura 13-2:** Adaptador Bluetooth CSR.  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

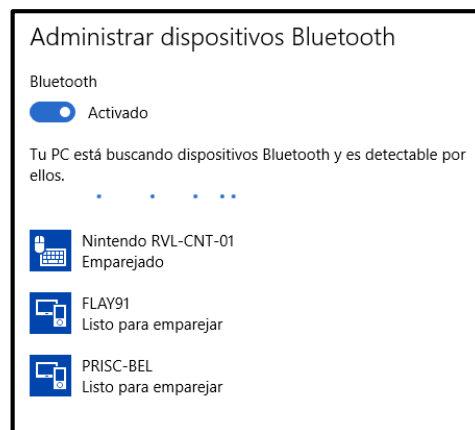


**Figura 14-2:** Instalación del controlador de Bluetooth  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

Luego se realiza el mismo proceso anterior, se busca el dispositivo y se vincula, no es necesario el ingreso de un código, se omite este paso y se prosigue normalmente.



**Figura 15-2:** Búsqueda del Mando Wii  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.



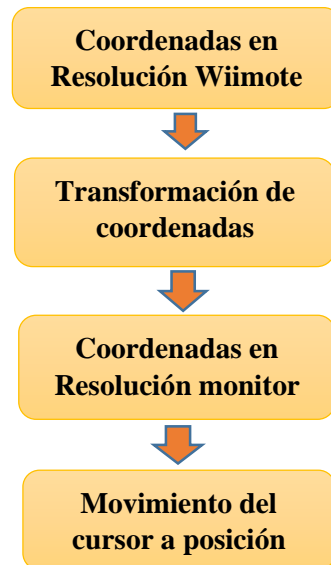
**Figura 16-2:** Emparejamiento del dispositivo.  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

#### 2.4.2.3 *Desarrollo de la Aplicación de la PDI.*

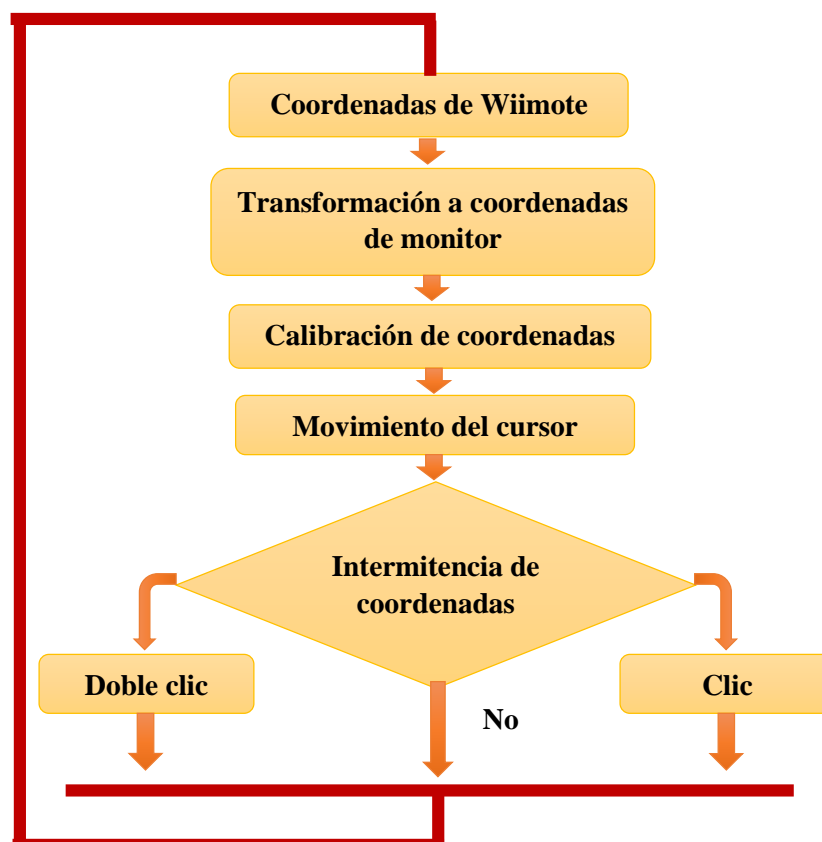
Esta aplicación se encarga de recoger la información enviada por el Wii Mando por parte del puntero infrarrojo, los mismos que son captados por su cámara, los interpreta y procesa identificando los datos que son utilizados para determinar la posición del puntero en la pantalla del computador y a su vez ejecutar algún tipo de acción.

Para el diseño y desarrollo de esta aplicación se utilizó el lenguaje de programación de alto nivel Visual Studio C# 2015, debido a sus características posee una gran variedad de herramientas que facilitan el desarrollo de aplicaciones, además soporta WiimoteLib que es una librería que permite usar el Wii Mando y aprovechar sus recursos.

Para el desarrollo de la aplicación se consideró los siguientes diagramas donde se detalla de forma general el procedimiento a seguir basado en dos procesos fundamentales como la calibración y el funcionamiento que debe seguir la PDI propuesta.



**Gráfico 1-2:** Proceso de calibración  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



**Gráfico 2-2:** Proceso de funcionamiento general  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

➤ **Inclusión del espacio de nombres.**

Para utilizar cualquiera de las clases de la librería dinámica es necesario incluir en el código fuente al espacio de nombres apropiado.

```
using WiimoteLib;
```

➤ **Referencia al Wiimote.**

El acceso a la información del control se realizará a través de esta referencia, motivo por el cual la utilizo como atributo de la instancia.

```
private Wiimote wm;
```

➤ **Instanciación de referencia con el Wiimote.**

Para que la creación de la instancia y posterior conexión sean exitosas es necesario que el control se encuentre ya asociado al equipo a través de la conexión Bluetooth.

```
try
{
    wm = new Wiimote();
}
catch (System.IO.IOException ioe)
{
    Console.WriteLine("No es posible ejecutar la aplicación: " + ioe.Message);
}
```

➤ **Registro de eventos.**

Se establecen métodos para manejar dos tipos de eventos del control:

```
wm.WiimoteChanged += wm_WiimoteChanged;
wm.WiimoteExtensionChanged += wm_WiimoteExtensionChanged;
```

➤ **Establecimiento de la conexión.**

```
try
{
    wm.Connect();
}
catch (Exception wnfe)
{
    Console.WriteLine("No es posible ejecutar la aplicación: " + wnfe.Message);
}
```

➤ **Consulta de la batería.**

```
int battery = (int) ws.Battery;
```

### ➤ Cierre De La Aplicación

El cierre de la aplicación puede también gestionarse como un evento. Así pues, para desconectar el Wiimote cuando la aplicación sea cerrada nos suscribiremos a este evento y llamaremos al método Disconnect cuándo éste sea lanzado.

```
private void Form1_Load(object sender, EventArgs e)
{
    FormClosing += ClosingApp;
}
void ClosingApp(object sender, FormClosingEventArgs closing)
{
    wm.Disconnect();
}
```

## 2.4.3 Desarrollo del control inalámbrico

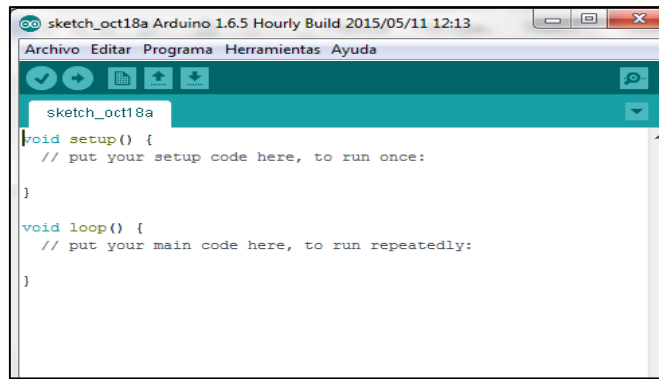
### 2.4.3.1 Conexión del Nunchuk con el Ordenador

Para manipular el nunchuk inalámbricamente procedemos a cambiar el medio de transmisión guiado (cable) a inalámbrico. La extensión del Wii Mando, el Nunchuk se comunica utilizando el protocolo de comunicación del bus I2C donde el intercambio de información se basa en una arquitectura maestro-esclavo.

Para la lectura e interpretación de los datos enviados por el nunchuk utilizaremos un Nano Arduino que es una plataforma de hardware de código abierto con un software fácil de usar, y a su vez, para la transmisión de datos de forma inalámbrica usaremos los módulos Xbee lo que permitirá ejecutar ordenes de cualquier lugar dentro del alcance del mismo.

#### 2.4.3.1.1 Instalación del Software Arduino

El software de Arduino nos permitirá realizar la programación respectiva en el módulo de Arduino Nano, la instalación de dicho software no es complicada los pasos a seguir son los siguientes: ejecuta la aplicación dando doble clic y seguimos aceptando las ventanas que nos aparecen dando clic en el botón de *next* o aceptar dependiendo el idioma hasta que queda instalado en su totalidad, luego simplemente ejecutamos la aplicación y nos muestra la pantalla principal.



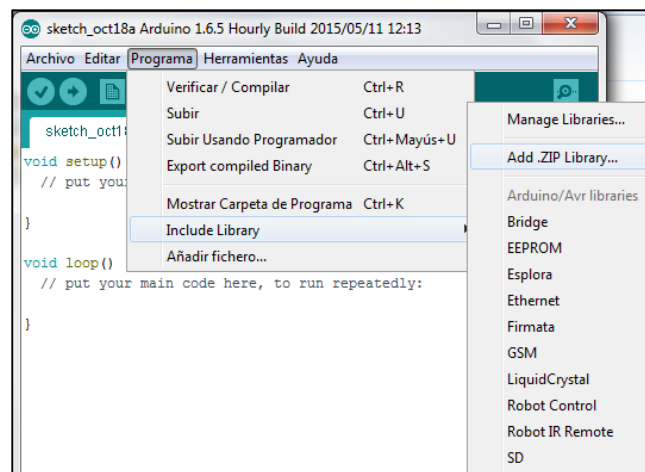
**Figura 17-2:** Instalación del software Arduino

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 2.4.3.1.2 Programación en el Nano Arduino

##### 2.4.3.1.2.1 Incorporación de la librería WiiChuck.h

La librería WiiChuck.h se utiliza para mostrar por el puerto serie los valores del mando. Se puede descargar directamente de la web de Arduino, para su incorporación simplemente se debe copiar el archivo descargado en la carpeta de archivos "libraries" del Software Arduino o añadir desde el programa.



**Figura 18-2:** Instalación de la librería WiiChuck.h

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

##### 2.4.3.1.2.2 Programación

Se incluyen librerías, las cuales realizan un gran trabajo ya que su función simplifica el código fuente al programador.

`#include "Wire.h"` : Reconoce el protocolo de comunicación I2C.  
`#include "WiiChuck.h"` : Reconoce al Nunchuk.

Iniciamos la comunicación serial y fijamos una velocidad en baudios, también necesitamos iniciar el Wiichuck y actualizar los valores constantemente.

```
Serial.begin(115200);  
chuck.begin();  
chuck.update();
```

Se utiliza una función de control que nos permita fijar condiciones y ejecutar alguna instrucción, tanto con los valores para el eje X como el eje Y.

```
If (chuck.readJoyX() != 0)  
If (chuck.readJoyY() != 0)
```

También debemos establecer condiciones, para los datos de los botones C Y Z.

```
if (chuck.buttonZ)  
if (chuck.buttonC)
```

Por último se imprime los valores enviados por los componentes Del Nunchuk.

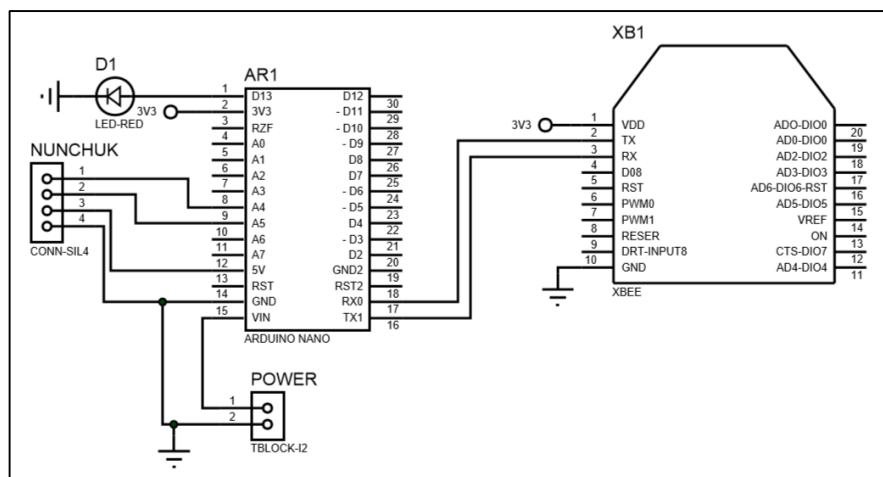
```
Serial.println(chuck.readJoyX());  
Serial.println(chuck.readJoyY());  
Serial.println("C");  
Serial.println("Z");
```

#### 2.4.3.1.3 Vinculación de los módulos Xbee.

##### 2.4.3.1.3.1 Conexión física del emisor/receptor

Como primer paso procedemos a cortar el revestimiento del cable del Nunchuk en el extremo del conector, dentro del cual encontramos cuatro hilos cuya función es la siguiente: positivo, negativo, datos y reloj los cuales conectamos en el módulo arduino en el siguiente orden:





**Figura 19-2: Conexión del Nunchuk, Arduino y Xbee**

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

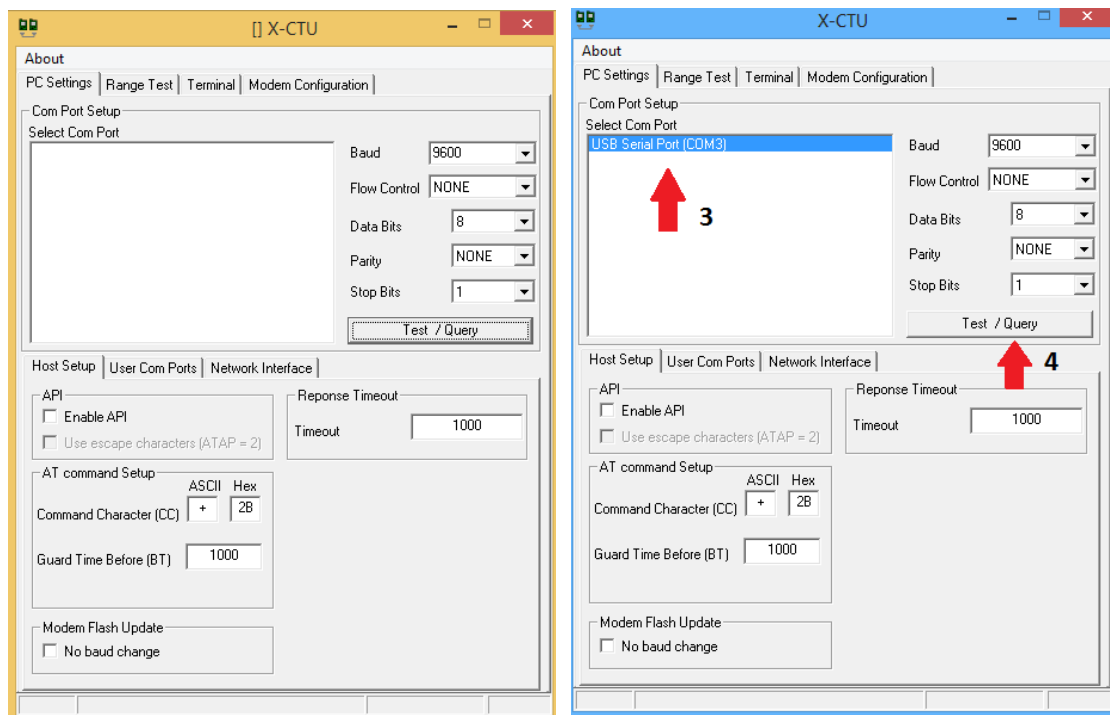
En el ping número 1(D13) del Arduino se polariza positivamente con 12v, el segundo ping(3V3) utilizamos como fuente de voltaje para el módulo Xbee, el octavo ping(A4) utilizamos para sincronización de reloj, el noveno ping(A5) utilizamos para la transmisión de datos, el ping decimotercero utilizamos como fuente de voltaje para alimentar el Nunchuk con 5V, el decimocuarto ping (GND) polariza negativamente al Arduino y al Nunchuk, el decimosexto y decimoséptimo ping utilizamos para la transmisión y recepción de datos respectivamente.

En el módulo Xbee utilizamos el primer ping (VDD) para alimentar con 3,3V, el segundo y tercer ping utilizamos para transmisión y recepción respectivamente y el décimo ping (GND) se utilizó para polarizar negativamente el módulo Xbee.

#### 2.4.3.1.3.2 Configuración del sistema

Con ayuda del programa XCTU realizamos la vinculación de los dispositivos, únicamente necesitamos los códigos que vienen impresos detrás de cada módulo, para vincular seguimos los siguientes pasos:

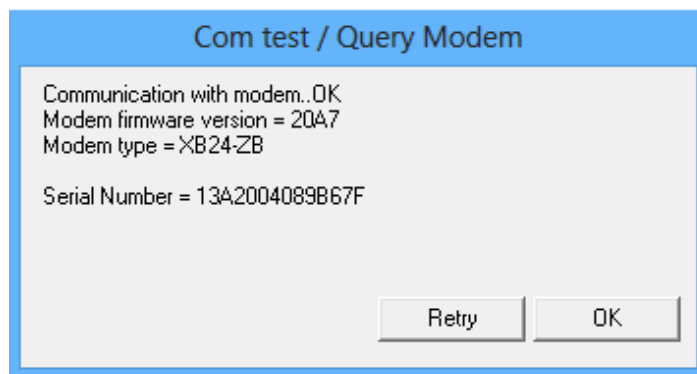
Ejecutamos XCTU:



**Figura 20-2:** Vinculación con X-CTU

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

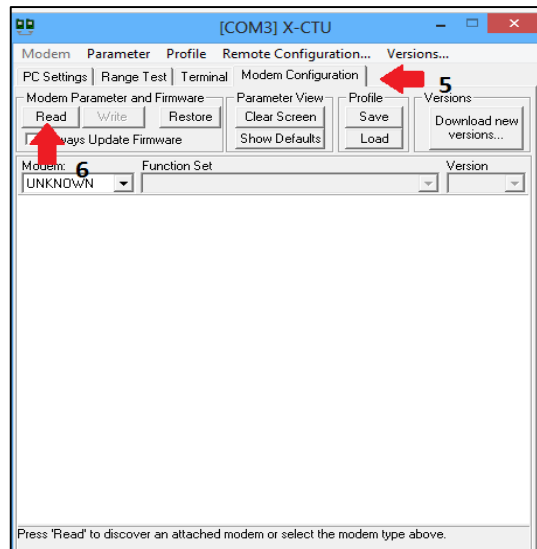
En la opción select com port y elegimos el COM de nuestro dispositivo, para verificar si efectivamente está funcionando damos clic en test query si nos visualiza las características de nuestro dispositivo presionamos Ok.



**Figura 21-2:** Verificación Test/Query

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

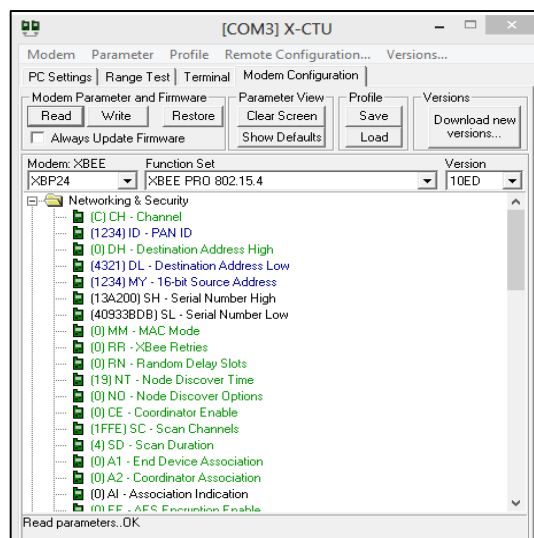
Conectamos el cable USB en el ordenador y su extremo en el adaptador del módulo Xbee, XCTU reconoce inmediatamente al dispositivo conectado, nos dirigimos a la opción Modem Configuration, luego damos clic en Read para que visualice automáticamente el dispositivo y podamos ingresar los códigos del Xbee con el cual se pretende vincular, en éste caso ingresamos el código del Xbee (XB2) es decir del módulo contrario.



**Figura 22-2: Función Read del XB1**

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

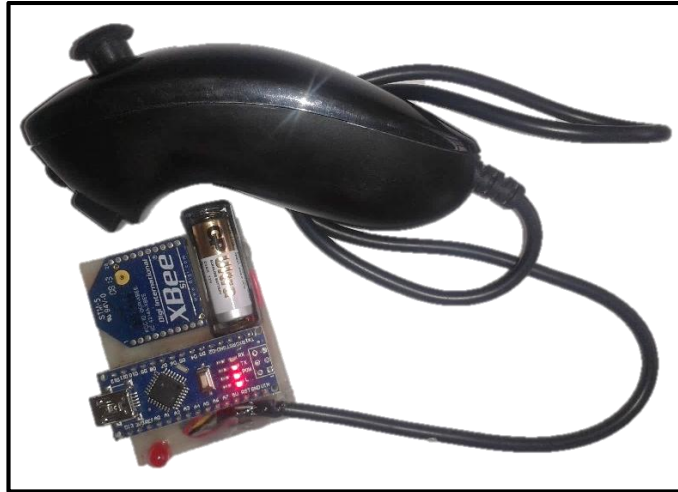
De igual forma conectamos el segundo Xbee en el adaptador e ingresamos el código del primer Xbee (XB1), al terminar éste proceso los dos módulos ya pueden comunicarse entre sí y pasar información inalámbricamente.



**Figura 23-2: Funcion Read del XB2**

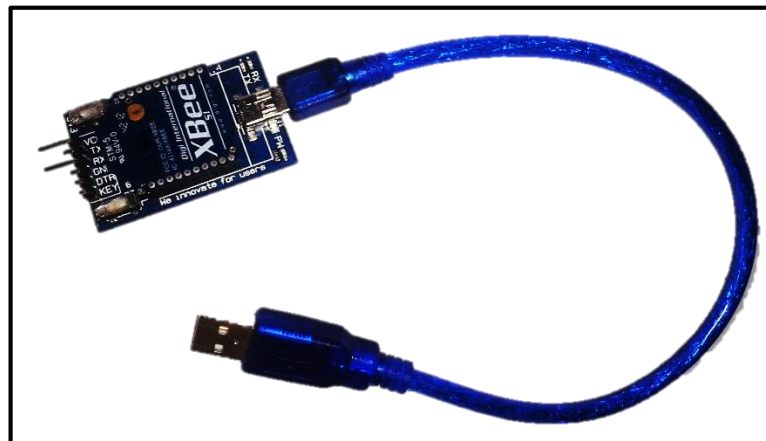
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

La comunicación con la cual interactúa el mando y el Nunchuk es i2c, al llegar al Arduino éste mediante un código de programación y la librería de Nunchuk con la que cuenta Arduino, esto ayuda a convertir la información de formato i2c a serial, luego esta información pasa al módulo Xbee (XB1) como muestra en la Figura 24-2 el cual envía inalámbricamente a su receptor (XB2) y finalmente llega al ordenador por cable serial siendo esta información procesada por el programa correspondiente.



**Figura 24-2: Modulo Emisor**  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

El segundo módulo Xbee (XB2) únicamente recibe la información es decir está actuando como un puente para que la información llegue al ordenador, la placa a la cual está conectado se encarga de convertir la los datos en formato serial.

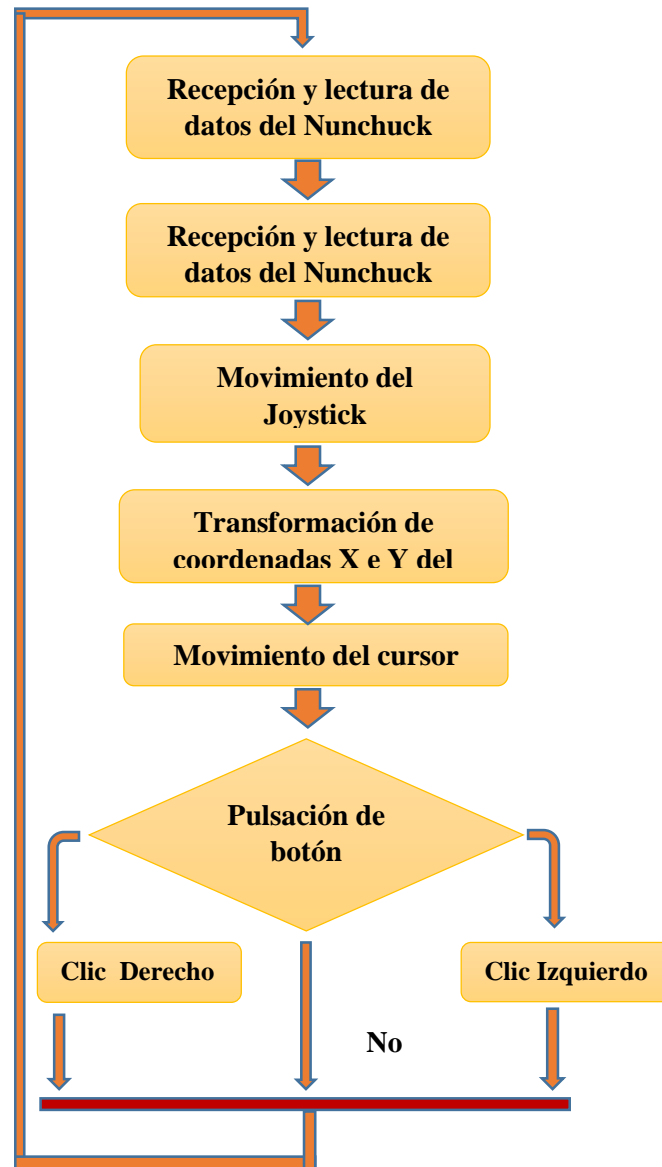


**Figura 25-2: Modulo Receptor**  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

#### 2.4.3.2 *Desarrollo de la Aplicación del Control Inalámbrico de la PDI*

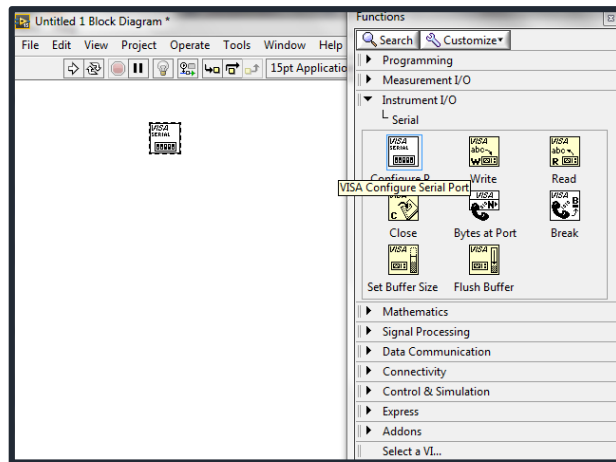
Se Utilizó el software LABVIEW para su desarrollo, mediante el cual se analizó y procesó la información recibida por el puerto serial, y utilizando procedimientos lógicos y matemáticos se elaboró una aplicación que permite controlar el indicador o cursor del computador a través del Nunchuk.

Dicho dispositivo como se detalla en el Capítulo I cuenta con varios componentes internos, de los cuales nos enfocaremos en el joystick y en los dos botones que presenta en su parte frontal, los mismos serán la base para el funcionamiento y mando del Control Inalámbrico de la PDI; el desarrollo de esta aplicación se llevó a cabo a través del siguiente proceso:



**Gráfico 3-2:** Proceso de calibración  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

Para utilizar los datos de la comunicación serial utilizaremos las herramientas que nos proporciona la librería VISA, dentro del diagrama de bloques vamos a View – Functions Palette – Instrument I/O – Serial, escogemos VISA Configure Serial Port.

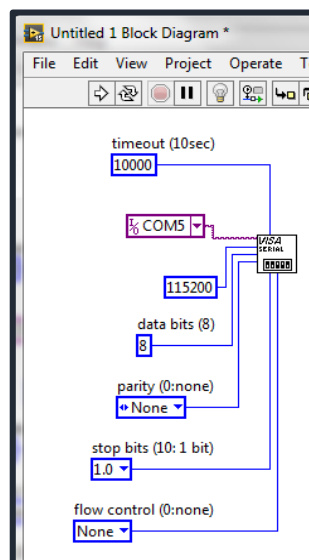


**Figura 26-2:** Interfaz de la herramienta VISA.

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Configuramos los parámetros necesarios dentro de Configure Serial Port que son necesarios para iniciar el funcionamiento y tener acceso del puerto serial de la PC, estos valores son:

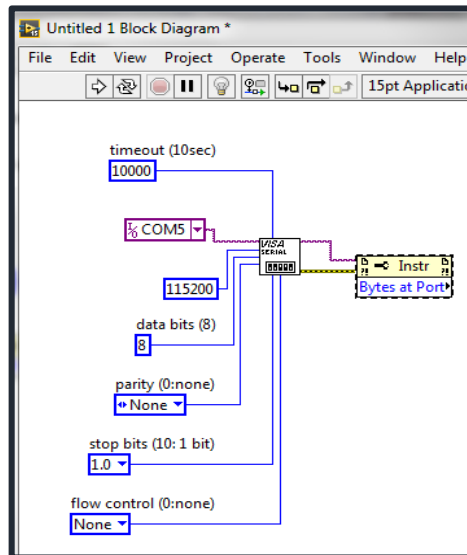
- El puerto que utilizaremos es el COM5.
- La tasa de transferencia será de 115200 baudios
- Los bits de datos comúnmente considerados son 8
- Los bits para el tiempo de parada es 1
- El tiempo de espera es de 10000 ms.
- Los valores para la paridad y el control del flujo no son considerados



**Figura 27-2:** Parámetros de VISA Configure Serial Port

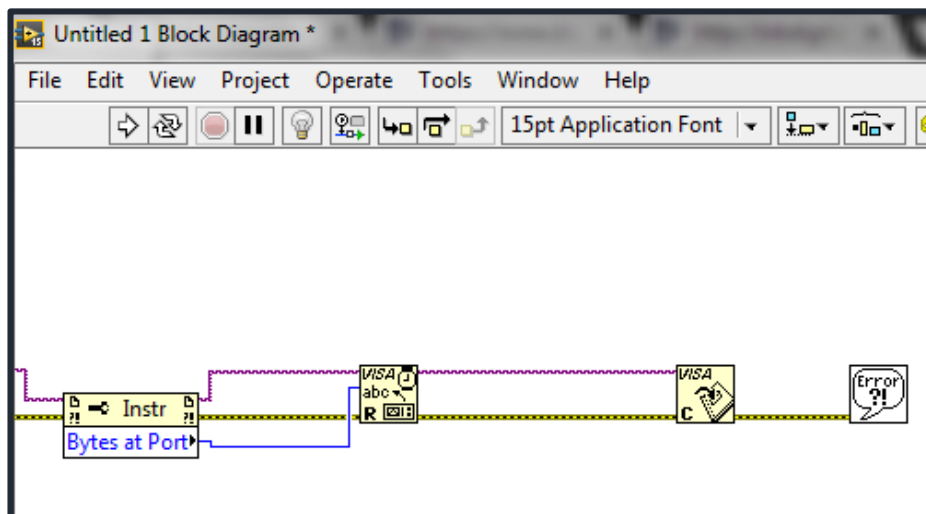
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Para manipular las propiedades de un objeto o aplicación utilizaremos la función VISA Property Node, el cual lo configuramos como Number of Bytes at Serial Port con el cual obtendremos toda la información del buffer es decir el número de bits que se encuentran en el puerto para ser leídos.



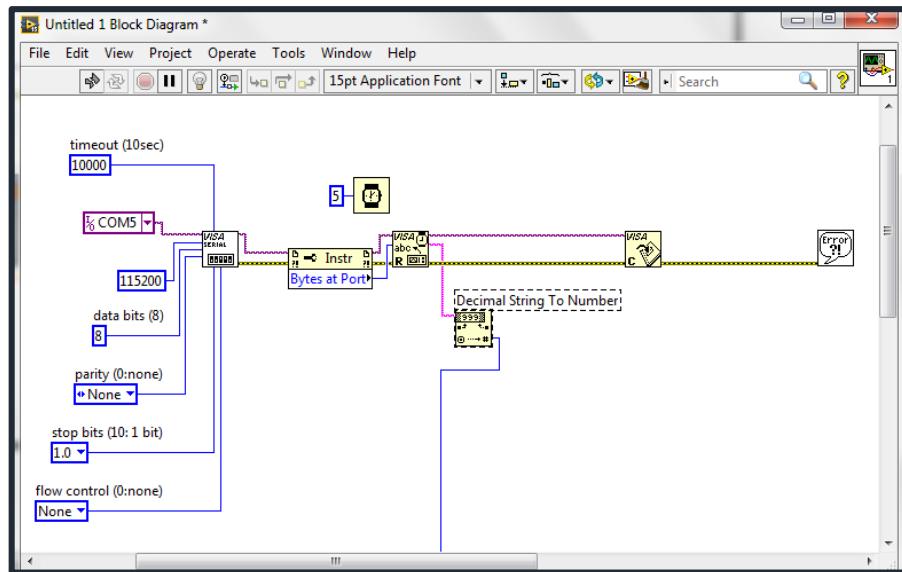
**Figura 28-2:** Unión de Visa Configure Serial Port y Property Node  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Para completar la sesión VISA utilizaremos un VISA Read y un VISA Close, el primero lee los datos enviados del Property Node y los almacena en un buffer virtual y el segundo cierra la sesión liberando el puerto, al final un controlador de errores denominado Simple Error Handler.



**Figura 29-2:** Sesión VISA  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

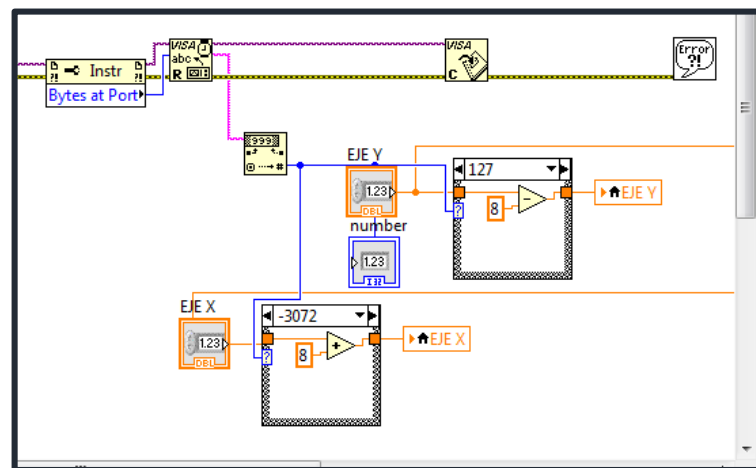
Para controlar el movimiento del cursor necesitaremos un Decimal String to Number para convertir la cadena de caracteres almacenados en el buffer del VISA Read.



**Figura 30-2:** Integración del Decimal String to Number

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Para determinar los movimientos en el eje X como en el eje Y del nunchuk, utilizaremos un Case Structure de tipo numérico y las funciones de Add y Subtract para establecer las condiciones necesarias que dicten su funcionamiento identificando los valores de las variables independientemente, se puede utilizar un indicador para visualizar los datos en el panel frontal.

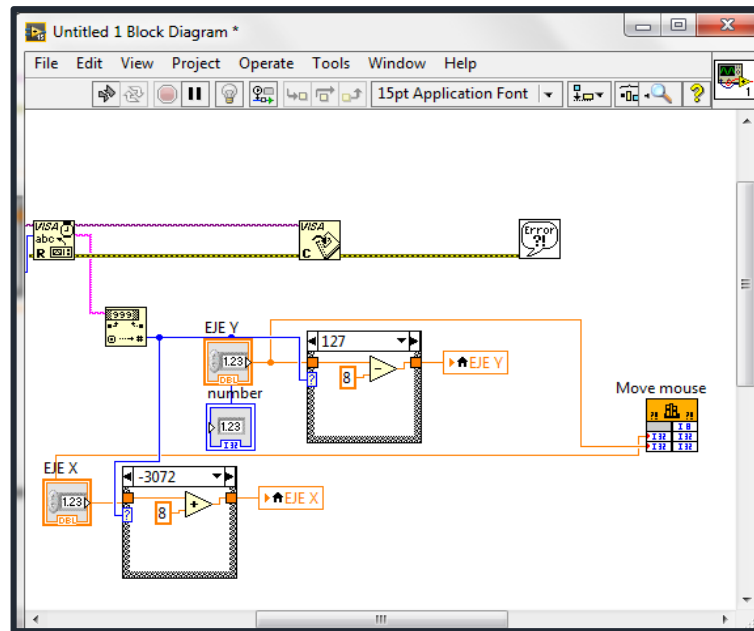


**Figura 31-2:** Case Structure para el eje X/Y

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

- El desplazamiento del cursor dependerá de esos valores, para generar dicha acción utilizaremos la función Call Library Function Node y la configuramos de acuerdo a nuestro propósito y enlazamos a la librería del sistema operativo que controla el movimiento del cursor.

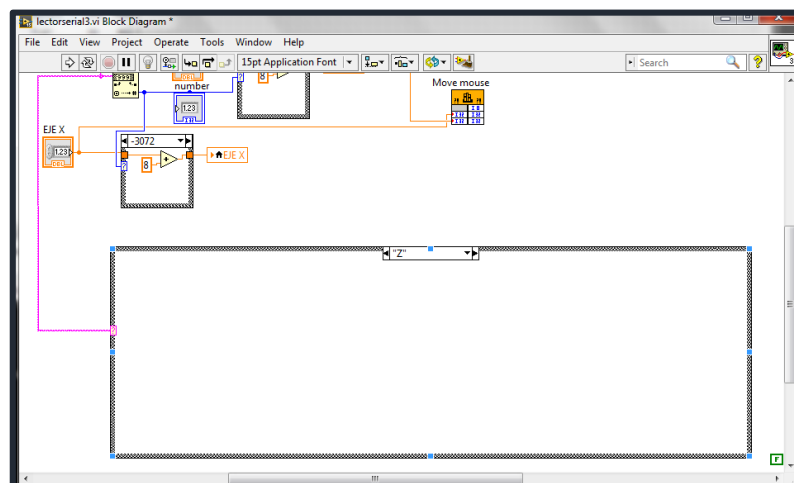




**Figura 32-2:** Incorporación de Call Library Function Node.

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

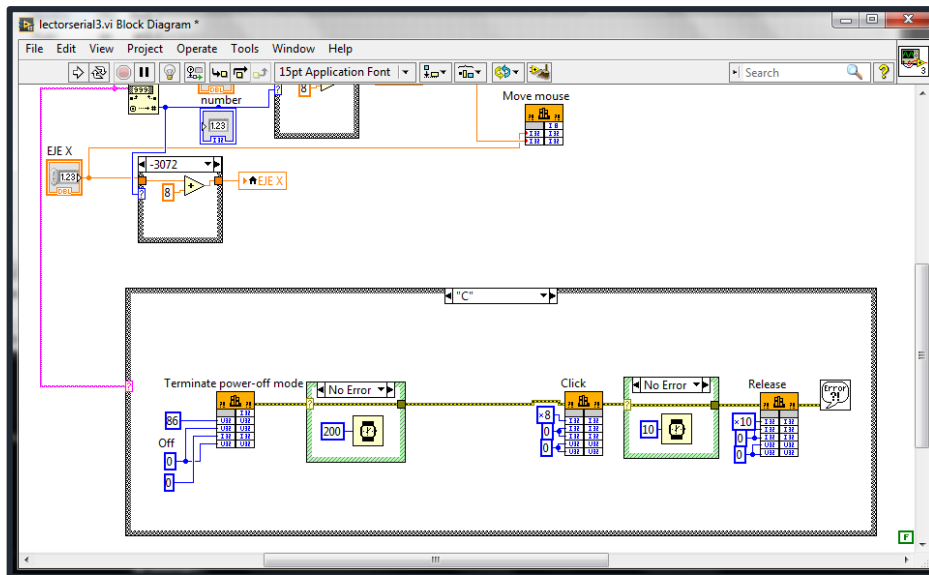
- Del mismo modo utilizaremos un Case Structure de tipo String para determinar las acciones de los botones C y Z, establecemos las condiciones y la acción que ejecutara el programa dependiendo el evento que se produzca.



**Figura 33-2:** Case Structure para los botones C y Z

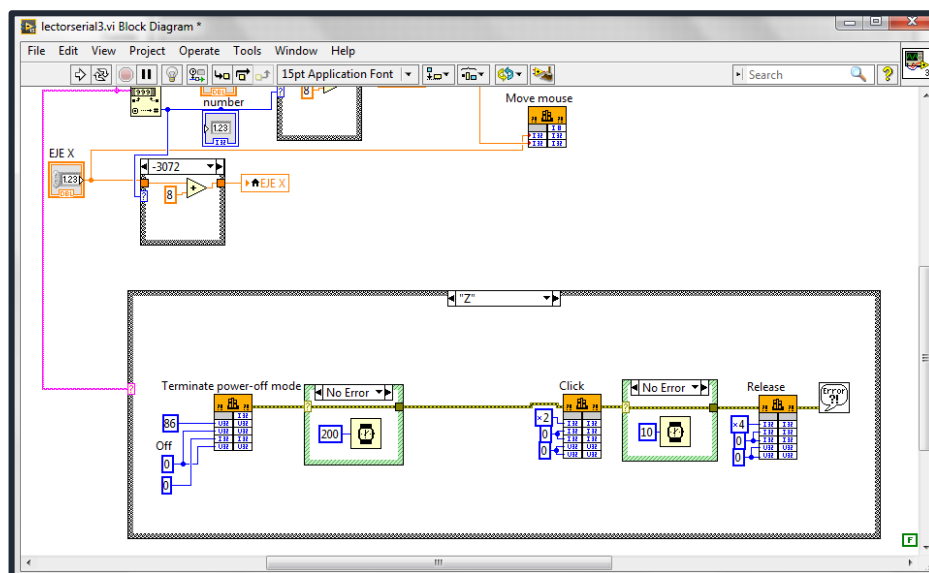
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Los botones del nunchuk realizaran la función de un clic de ratón izquierdo o derecho, dependiendo de los parámetros que se configuren en la función Call Library Function Node y las condiciones establecidas en el programa, además utilizaremos nuevamente un Case Structure y un tiempo de espera Wait.



**Figura 34-2:** Configuración para el evento C

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

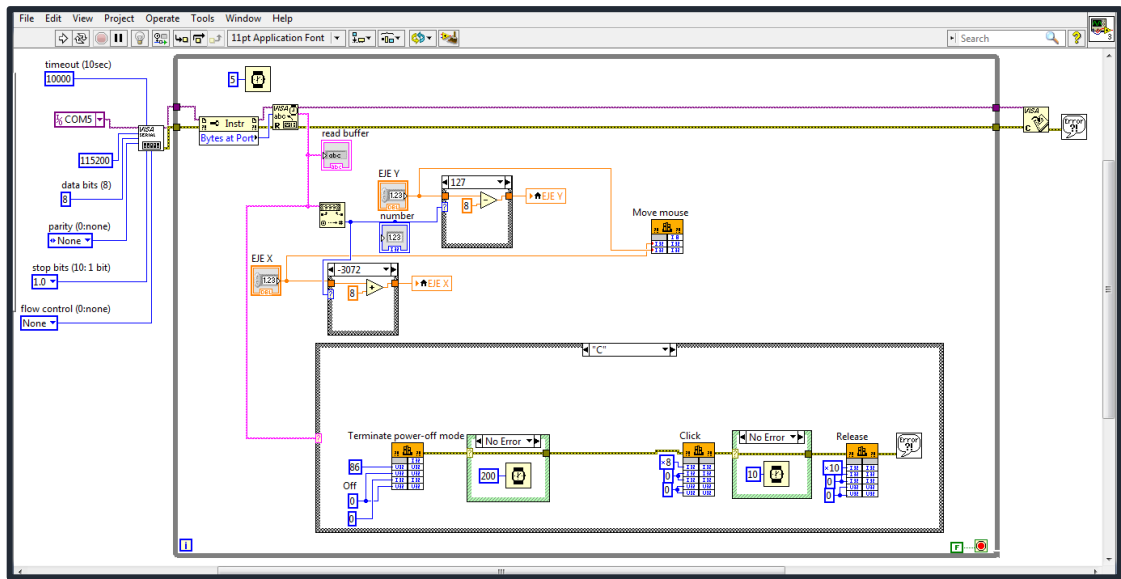


**Figura 35-2** Configuración para el evento Z

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

En el desarrollo del programa utilizaremos una instrucción de repetición denominada While Loop y una instrucción Wait con la cual el proceso del programa se ejecutara hasta que se cumpla o deje de cumplirse una condición establecida o se detenga por medio de otra instrucción.

Todos los elementos mencionados anteriormente se utilizaran de acuerdo a su función y las veces que sean necesarias en el desarrollo de la aplicación hasta obtener el funcionamiento requerido para el control inalámbrico.

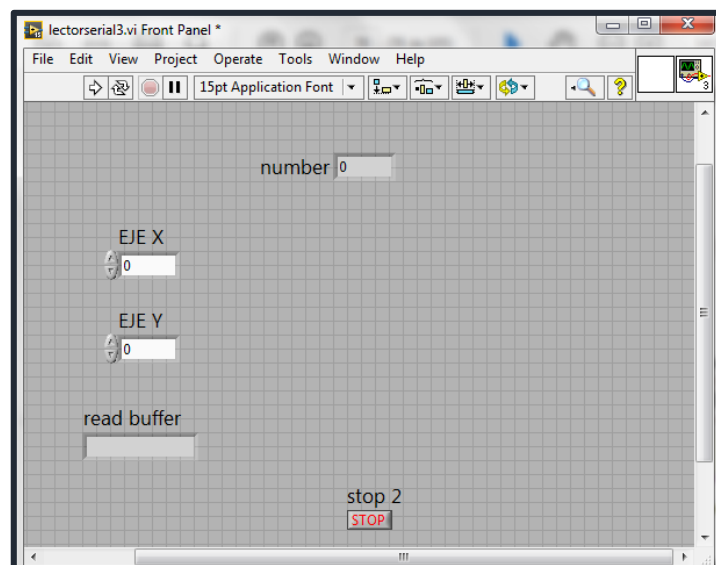


**Figura 36-2:** Integración de la instrucción While Loop.

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Para la verificación, comprobación y corrección de errores utilizaremos ciertos elementos para crear un interfaz gráfica en el panel frontal, donde se podrá visualizar valores de entradas, salidas y resultados de los procesos realizados en el diagrama de bloques, principalmente consta de:

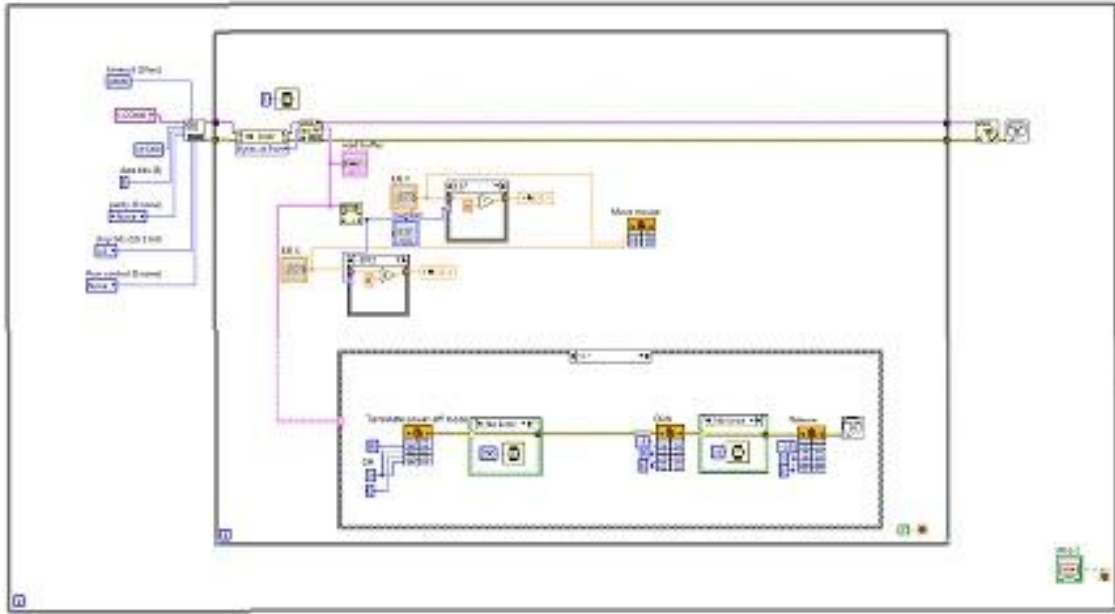
- Controles numéricos para el eje X y el eje Y.
- Un Indicador numérico para la salida del Decimal String to Number
- Un indicador string para el buffer
- Un botón para detener el programa.



**Figura 37-2:** Interfaz Gráfica para verificación de errores

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Esta aplicación no necesita una Interfaz gráfica muy elaborada o detallada, solo lo necesario para comprobar que la información que se está manejando sea la correcta y necesaria para cumplir con nuestro propósito, todo el proceso se realiza en el diagrama de bloques. En la Figura. 38-2. Se muestra la programación final del Control Inalámbrico de la PDI.



**Figura 38-2:** Programación del control inalámbrico

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 2.4.4 Desarrollo de la Aplicación Final

En esta etapa integraremos las aplicaciones de la Pantalla Digital Interactiva y la del Control inalámbrico en un solo programa, ambas aplicaciones fueron desarrolladas en diferentes plataformas para evitar problemas de funcionamiento al momento de enviar y recibir información. Al funcionar independientemente se evita este inconveniente y reduce un poco la dificultad del manejo tanto del Wii mando como del Nunchuk.

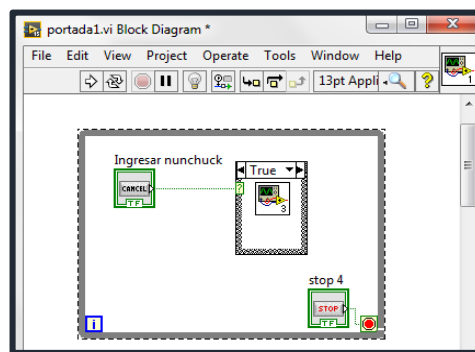
Sin embargo, es necesario crear una aplicación que pueda unir los dos programas logrando un funcionamiento simultáneo lo cual resulta menos complicado que al realizar todo el programa en una sola plataforma, logrando optimizar este recurso con el fin de aportar para el desarrollo de nuevas metodologías que garanticen un proceso educativo más dinámico e interactivo.

#### 2.4.4.1 Integración entre las Aplicaciones de la PDI y el Control Inalámbrico.

Utilizaremos nuevamente el programa anterior ya que presenta varias herramientas que permite su incorporación con otras plataformas, una de ellas es la plataforma .NET en la cual está basada la aplicación de la PDI.

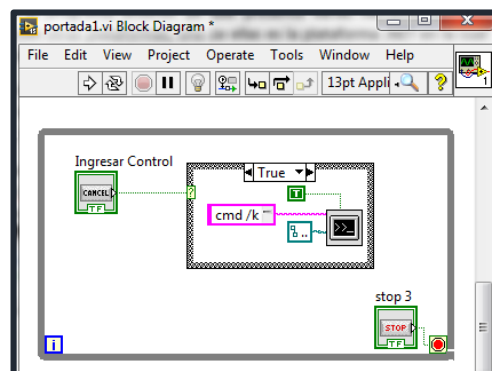
Creemos un nuevo programa, el cual se basara en tres sencillas etapas. Para el desarrollo del mismo por facilidad es recomendable que ambos programas se encuentren en la misma ubicación, el proceso es el siguiente:

- Para la primera parte utilizaremos un While Loop de propiedad booleana, en el cual se ejecutara un Case Structure que contiene el programa del Control inalámbrico, este se ejecutara y se detendrá hasta que exista un cambio en el estado de un pulsador.



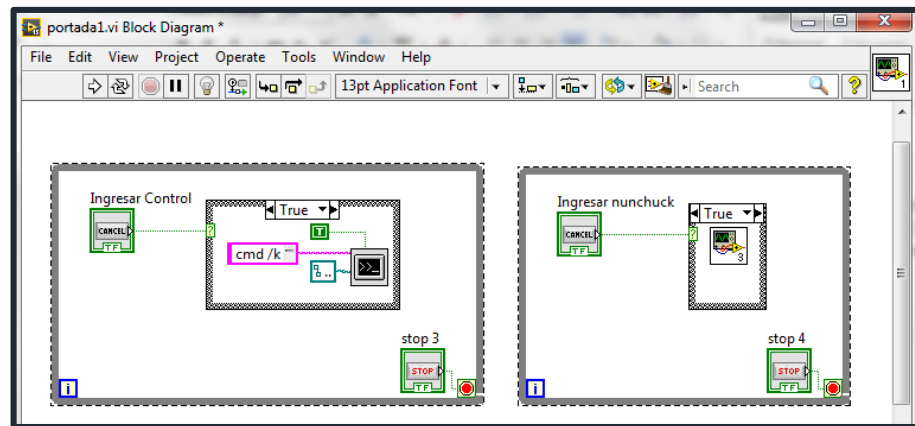
**Figura 39-2:** Vinculación con la aplicación de Control Inalámbrico  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

- En la segunda etapa, el procedimiento es similar con la diferencia de que se incluye la función de System Exec.vi con el cual una vez configurada (ver ANEXO B) podremos llamar a la aplicación de la PDI y se ejecutara de acuerdo a las condiciones establecidas.



**Figura 40-2:** Vinculación con la aplicación de la PDI  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

La integración de ambos sistemas en un solo programa, quedaría de la siguiente forma en el diagrama de bloques.



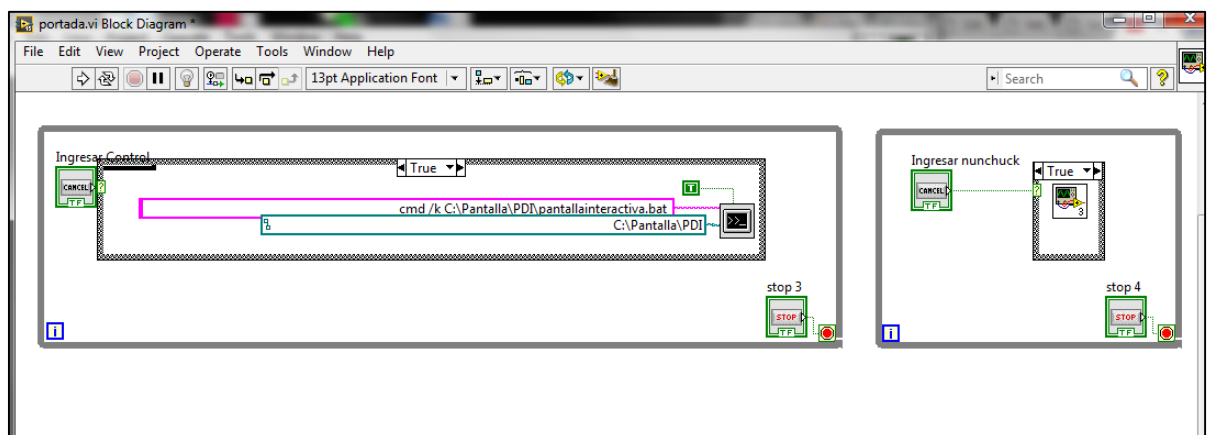
**Figura 41-2:** Programación de aplicación final.

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

#### 2.4.4.2 Generación de la aplicación ejecutable

Otra manera de utilizar estas aplicaciones es generando un archivo ejecutable del programa principal, para ello es necesario hacer modificaciones dentro del panel de bloques de su programación, para ello realizamos lo siguiente:

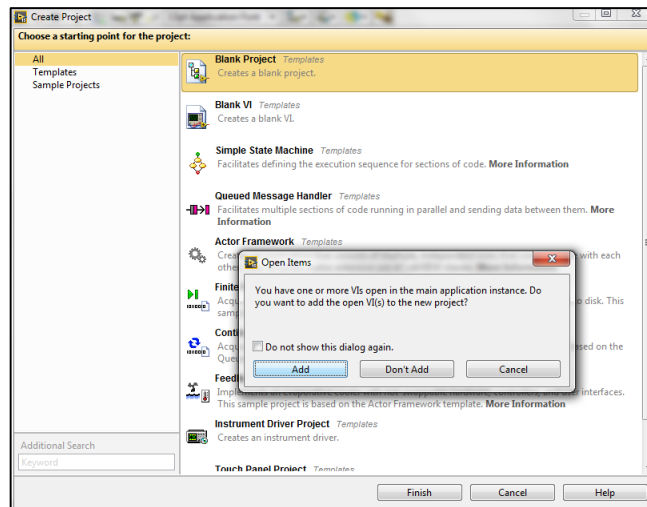
Se ingresa al bloque donde se realiza la vinculación con el archivo ejecutable de la Pantalla Interactiva y se modifica la dirección de enlace, se direcciona directamente al disco local C y se agregó un campo adicional para que se ejecute directamente.



**Figura 42-2** Modificación del programa principal

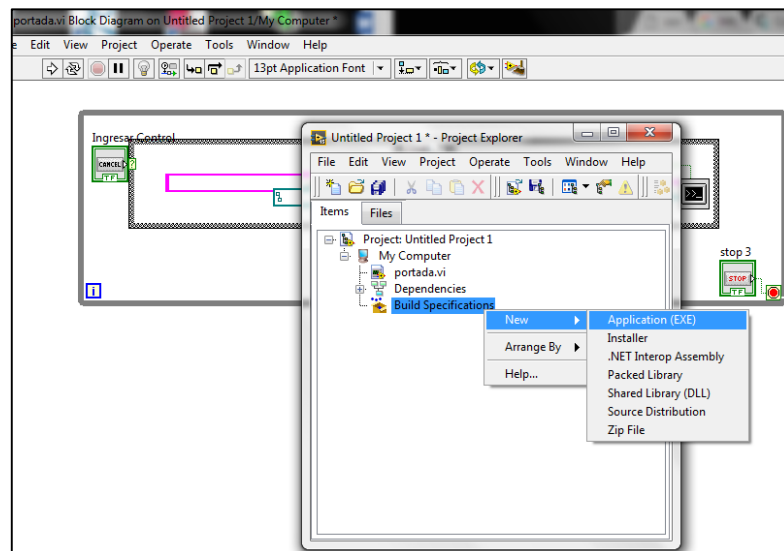
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Para lo cual se procede a crear un nuevo proyecto, en el cual se realizarán las modificaciones respectivas para cumplir con el objetivo propuesto.



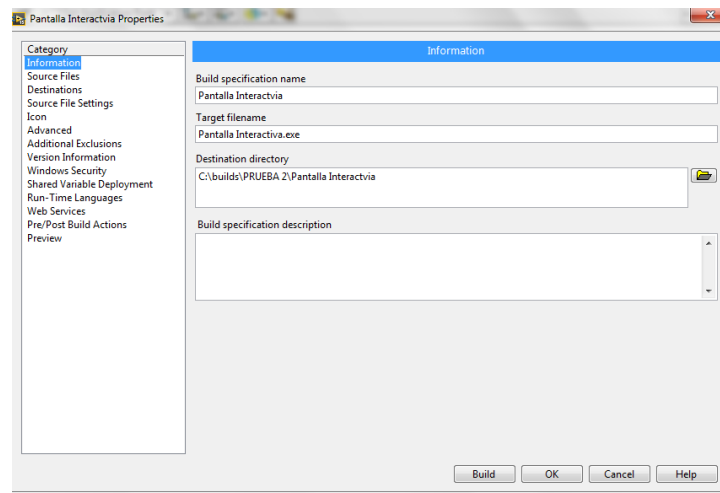
**Figura 43-2:** Creación de un nuevo proyecto  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Buscamos la opción que nos permita crear una aplicación ejecutable, la misma se encuentra en la opción “Build Specifications” como se muestra a continuación.



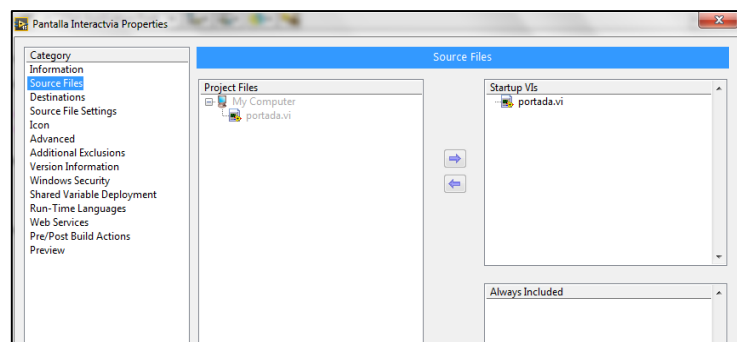
**Figura 44-2:** Configuración para una aplicación .EXE  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Se establece los parámetros de información donde designamos el nombre de la carpeta y del archivo con que se va a generar, también se designa la ubicación donde este se alojara.

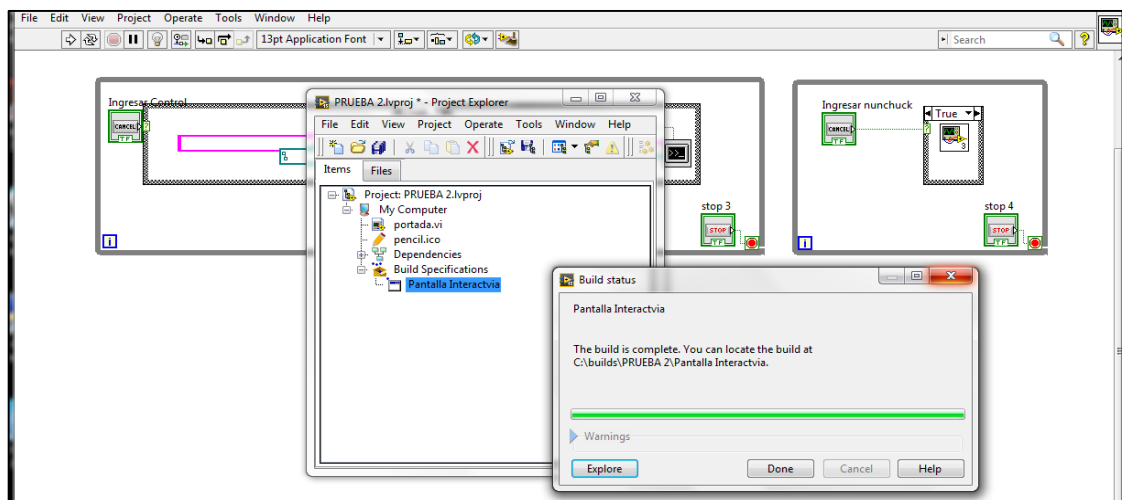


**Figura 45-2:** Configuración de parámetros de información  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Posteriormente vinculamos el archivo principal del programa con el que estamos creando, esto se realiza en la sección de “Source Files”

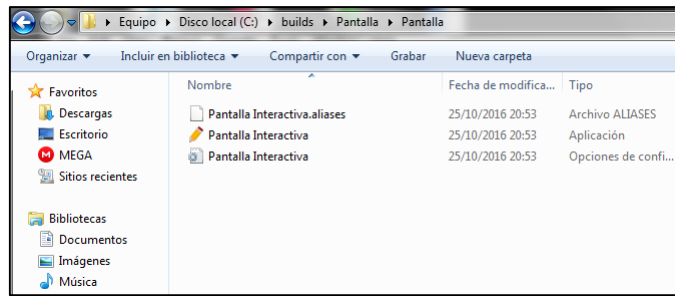


**Figura 46-2:** Vinculación con programa principal  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



**Figura 47-2:** Proceso para guardar archivo ejecutable  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.





**Figura 48-2:** Archivo ejecutable  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

#### 2.4.4.3 Interfaz gráfica

La aplicación de la Pizarra Digital Interactiva con Control Inalámbrico no contiene variables de entrada que se puedan ingresar manualmente ni acciones con las que se pueda interactuar desde el programa como opciones, tablas o gráficas, ya que no es necesario.

Todas las acciones se las ejecuta directamente desde el dispositivo Nunchuk o con el puntero infrarrojo, permitiendo al usuario interactuar con la PDI desde cualquier lugar dentro del rango establecido sin necesidad de manipular el computador una vez ejecutado el programa.

Esta aplicación consta de dos botones con los cuales podemos dar inicio al sistema que se requiera utilizar además dichos sistemas se encuentran gráficamente muy bien representados, para cambiar de programa solamente se debe seleccionar la opción restante y automáticamente cambiara de función.



**Figura 49-2:** Interfaz Gráfica de la Aplicación  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

## **CAPITULO III**

### **3 MARCO DE RESULTADOS, ANÁLISIS Y VALIDACIÓN**

#### **3.1 Introducción**

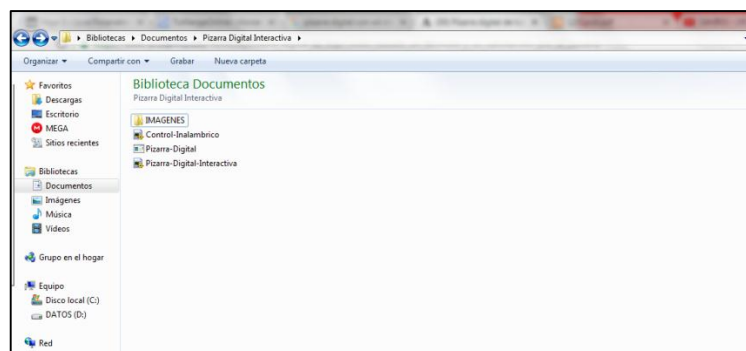
En el presente capítulo se determinará la eficiencia y el adecuado funcionamiento del prototipo obteniendo así los resultados anhelados propuestos anteriormente, para llevar a cabo se realizará la implementación del proyecto el cual consta de tres etapas; la pantalla interactiva, el control inalámbrico y la aplicación que se encarga de recibir analizar y ejecutar las peticiones del usuario.

#### **3.2 Instalación del software de la PDI y Control Inalámbrico**

El programa final de la Pantalla Digital Interactiva con Control Inalámbrico consta de tres archivos, uno de ellos es el principal que une las aplicaciones de los dos sistemas, sin embargo deben estar los tres archivos presentes para que se ejecuten sin ningún problema.

Para su instalación se copia la carpeta que contiene dichos archivos y se pega donde se desee guardar pero se debe configurar su ubicación, la aplicación esta direccionada por defecto para el disco local C en Windows, por lo cual se recomienda pegarlo en este lugar.

Para cambiar de ubicación a los archivos es necesario modificar la aplicación principal debido a que necesita conocer la dirección actual dentro del computador donde van a ser ubicados para poder enlazarlos, esta acción se realizó en el al final del Capítulo II.



**Figura 1-3:** Ubicación de archivos  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

### 3.3 Funcionamiento del sistema de la pantalla digital interactiva

El diseño propuesto para el desarrollo del presente sistema consta de dos etapas, las cuales se han dado a conocer en los capítulos anteriores, estos son la Pantalla Interactiva y el Control Inalámbrico, el funcionamiento de ambos componentes está determinado por el programa de control principal cuyo desarrollo se dio a conocer al final del capítulo anterior y en el apartado siguiente se explicara detalladamente su instalación y su manipulación.



**Figura 2-3:** Etapas de aplicación  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

El programa principal se encarga de controlar el funcionamiento de cada sistema de forma independiente pero integrándolos en una sola aplicación sin presentar problemas e inconvenientes en su rendimiento y función, logrando complementarse entre ellos en relación a su propósito el cual es ser utilizado como un recurso didáctico interactivo útil para metodologías y procesos educativos.

Su interfaz gráfica consta de dos opciones con las cuales se puede elegir el sistema que se desea utilizar, una vez que se ha iniciado un sistema se lo puede ejecutar el tiempo que sea necesario

sin presentar alteraciones en su funcionamiento siempre que se tomen en cuenta las indicaciones que se detallaran más adelante; para cambiar de opción solo es necesario presionar el botón de "ingresar" y automáticamente se ejecuta el otro sistema.

### **3.3.1 *Funcionamiento de la Pantalla Interactiva***

La opción de la Pantalla Interactiva, cuenta con la función de manipular el computador desde una pizarra común, para realizar esta acción en la pizarra se debe visualizar la imagen de la misma por medio de un proyector y mediante el uso un puntero especial realizar acciones como seleccionar, arrastrar, instrucción de clic izquierdo y doble clic.

Para su correcto funcionamiento es necesario definir el área donde se va a trabajar, por lo cual se incorporó una ventana adicional donde aparece la opción de "calibración" que permite seleccionar el espacio que se va a utilizar; esta opción se muestra al iniciar el programa de la PDI, además cuenta con un indicador de muestra el estado de las baterías que utiliza el dispositivo para indicar cuando es necesario un cambio de las mismas.



**Figura 3-3:** Opción de calibración

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

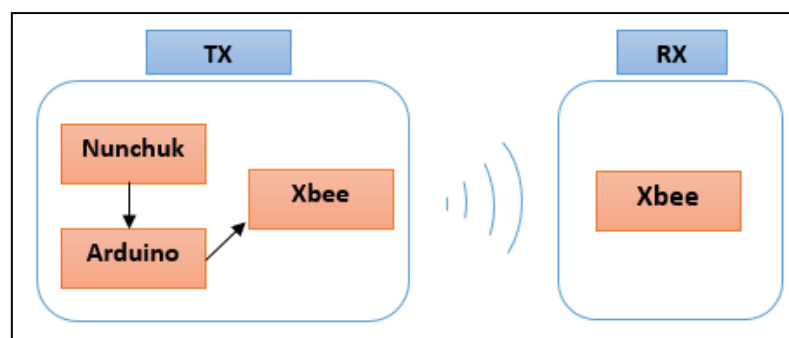
Todas estas acciones que realice el usuario son controladas, procesadas y ejecutadas por el programa desarrollado con ayuda del dispositivo necesario (Wii Mando), de tal forma que la persona no necesite realizar alguna configuración especial o adicional para su utilización.

### **3.3.2 *Funcionamiento del Control Inalámbrico***

La opción del Control Inalámbrico, cuenta con las mismas funciones de la PDI con la inclusión del clic derecho y diferencia destacable la cual radica en su utilización, este control puede funcionar desde cualquier lugar dentro un rango establecido el cual se detallara más adelante, evitando estar cerca del ordenador todo el tiempo.

El Control Inalámbrico se encarga de manipular el cursor de la PC, debido a sus componentes físicos como el joystick y botones se puede recrear eficientemente los movimientos y operaciones del mismo, el control envía instrucciones de acuerdo a la acción que se realice en el dispositivo, su respectiva aplicación recibe esta información la procesa y ejecuta dentro del sistema la operación que se desea realizar.

Para dar inicio al sistema, primeramente se debe conectar el dispositivo para la recepción de datos (Modulo Xbee) para luego simplemente seleccionar la opción correspondiente dentro del programa principal, debido a que tampoco necesita parámetros de configuración de forma manual no es necesario más opciones dentro de la interfaz gráfica de su programa.



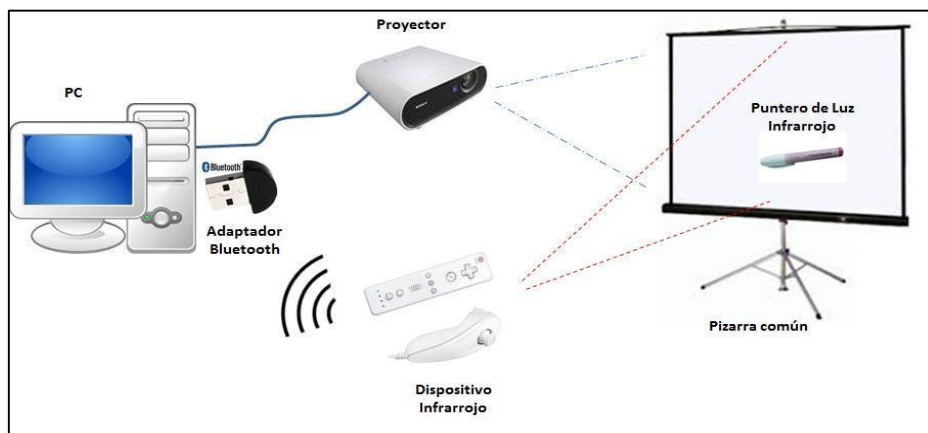
**Figura 4-3:** Diagrama de bloques del control inalámbrico  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

### 3.4 Implementación

Para realizar la implementación del prototipo se requieren los siguientes componentes: un Proyector, una superficie plana, un adaptador Bluetooth en caso de tener incorporado en el ordenador no hace falta el adaptador, un Mando Wii, un Nunchuk y un puntero infrarrojo. En Figura 5-3 se puede apreciar el escenario de forma más intuitiva para una mejor comprensión.

Además se debe considerar el espacio donde se va utilizar, es decir se debe tomar en cuenta el tamaño de la superficie donde se desea proyectar la imagen la cual debe estar dentro de las dimensiones del alcance de proyección del Mando Wii, además de la altura y los ángulos a los cuales se debe colocar el dispositivo para que su funcionamiento sea adecuado.

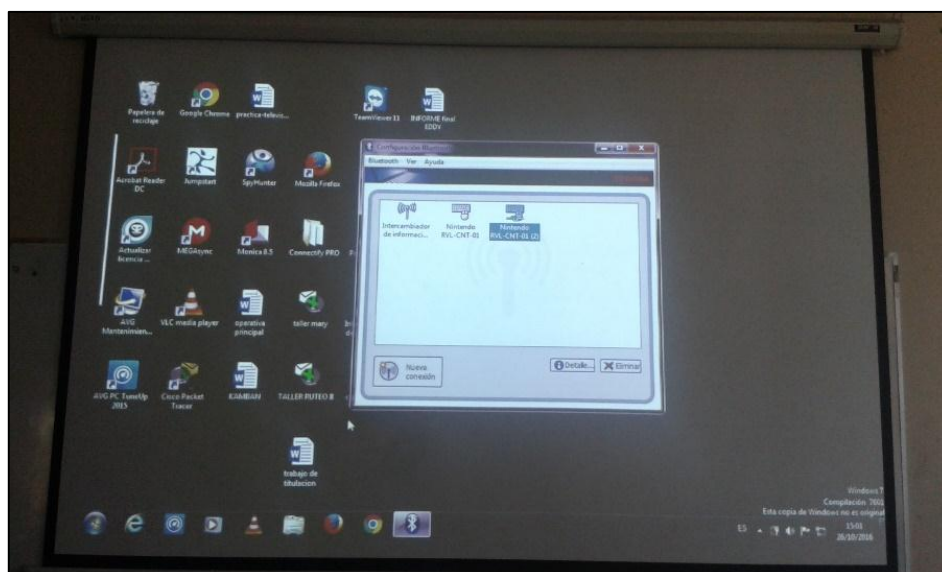
Los detalles para una correcta implementación y uso con las mejores prestaciones del sistema se indican de forma detallada desde la vinculación del dispositivo, su funcionamiento y limitantes.



**Figura 5-3:** Esquema del Prototipo  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

### 3.4.1 Vinculación del Mando con el ordenador

Si el ordenador no cuenta con Bluetooth incorporado, conectar un adaptador Bluetooth e instalar el driver correspondiente para que la PC reconozca el adaptador, luego de instalarlo elegimos la opción de “buscar dispositivo Bluetooth”, en el mando viene incorporado un pulsador el cual se ubica debajo de la tapa de las baterías, presionamos el pulsador por 2 segundos y soltamos, a partir de este momento el Mando se pone en modo visible y puede ser detectado por otro dispositivo para su vinculación.



**Figura 6-3:** Vinculación Bluetooth  
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

### 3.4.2 Manipulación del software

Para abrir el archivo ejecutable simplemente damos doble clic sobre el archivo y nos muestra una interfaz como la que podemos apreciar en Figura 7-3, nos visualiza dos opciones la primera de ellas nos permite interactuar directamente con la imagen proyectada en la pizarra, es necesario recordar que debemos establecer el área en la que vamos a trabajar.



**Figura 7-3:** Portada de la aplicación

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

El proceso de calibración es indispensable dentro de la ejecución del programa ya que garantiza su buen funcionamiento y la precisión, que es fundamental en el proceso interactivo para que las acciones efectuadas por el usuario en la pizarra sean las mismas que se ejecuten en el ordenador, es decir la acción se debe dar en el mismo lugar y aproximadamente al mismo tiempo según el requerimiento del usuario.

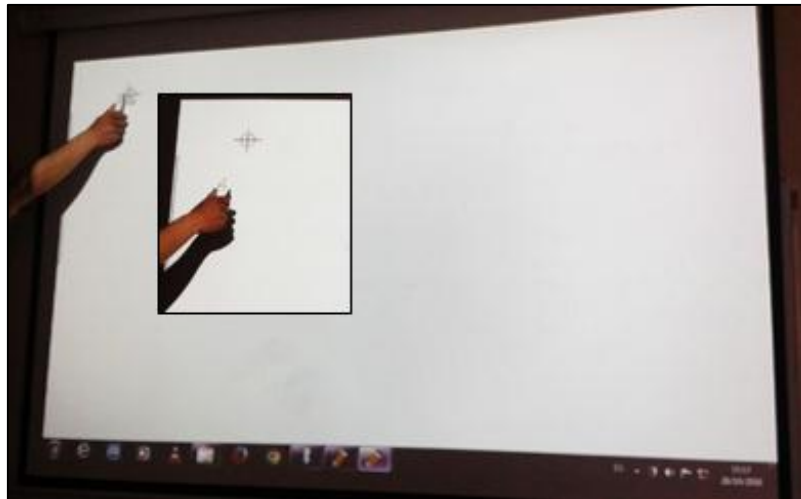


**Figura 8-3:** Ejecución de la aplicación de la PDI

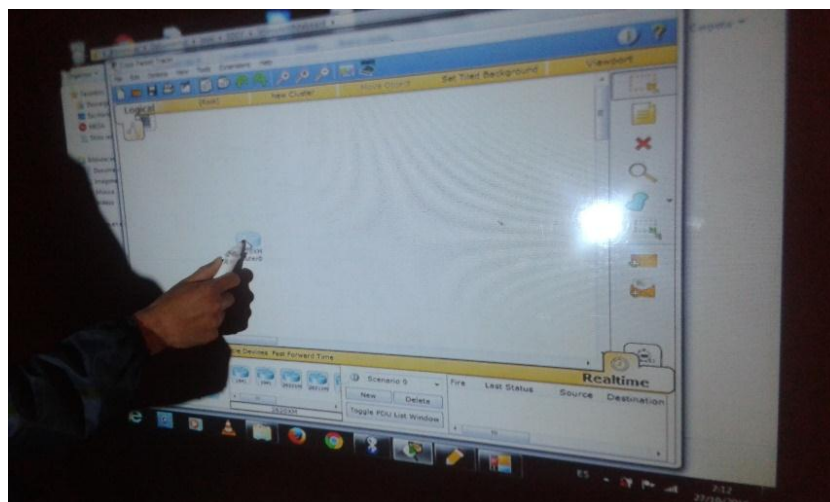
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



En caso de mover el Mando Wii o el proyector es necesario volver a calibrar la pantalla debido que empieza a reducir la precisión, para llevar a cabo la calibración se debe colocar el puntero infrarrojo en cada esquina empezando por la parte superior izquierda Figura 9-3, seguido de la esquina superior derecha luego en la esquina inferior izquierda y finalmente en la esquina inferior derecha.



**Figura 9-3:** Calibración de la PDI  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.



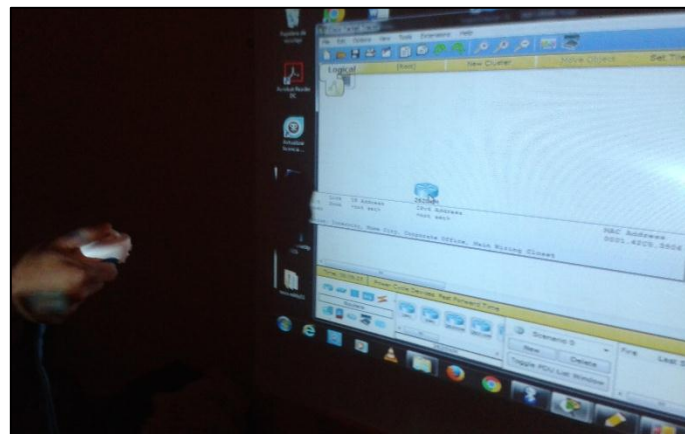
**Figura 10-3:** Manipulación de la PDI  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

La segunda opción es la de activar el modulo del nunchuk el cual actúa como un mouse inalámbrico, tomando el control de la pantalla a una cierta distancia, dinamizando más el entorno interactivo, para la implementación del dispositivo se debe conectar el arduino nano a un puerto USB del ordenador luego debe ejecutar la aplicación nunchuk y ya puede empezar a utilizar.





**Figura 11-3:** Ejecución de la aplicación del control inalámbrico  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.



**Figura 12-3** Manipulación de control inalámbrico  
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

### 3.5 Resultados

Después del proceso de diseño y desarrollo de la pantalla digital y del software para controlar el Mando Wii y el Nunchuk, el siguiente paso es verificar su correcto funcionamiento con las diferentes pruebas a realizar.

Las pruebas se realizaron en:

- La FIE-EIE-TR                      -1.654013, -78.677351
- Los Olivos                            -1.657566, -78.681953



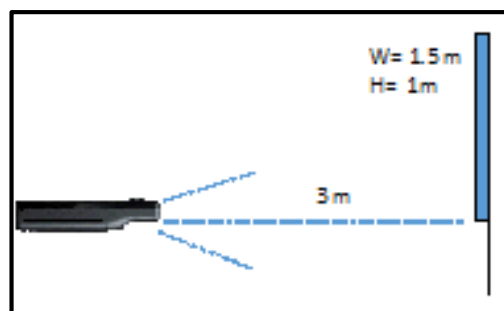
**Figura 13-3:** Lugares de prueba para el sistema

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Se tomó estos dos puntos de la ciudad de Riobamba debido a la facilidad del espacio requerido y los elementos necesarios para las pruebas pertinentes con el fin de obtener todos los datos necesarios para determinar el comportamiento del sistema y detalles técnicos que se consideraran para su posterior implementación.

### 3.5.1 Determinación de la ubicación del Mando Wii

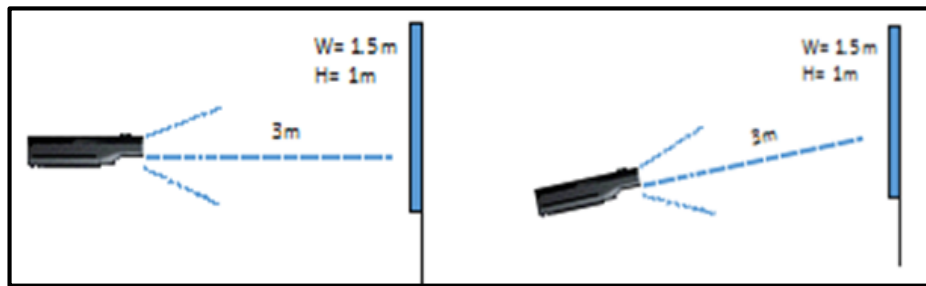
Conociendo el rango de distancia a la cual debe ubicarse el Wii Mando para que funcione correctamente (de 3 a 10 pies) del área donde se proyecta la imagen, calcularemos la altura necesaria a la que debe colocarse para que la cámara puede cubrir una pizarra con una dimensión de 1.5m de ancho y 1m de largo.



**Figura 14-3:** Calculo de distancia para el Mando Wii

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Según el cálculo realizado (Anexo A), el Wii mando debe ubicarse a 40 cm más arriba del nivel de la pizarra en línea recta, o a su vez debería elevarse aproximadamente 8 grados sin moverlo de su posición, para esto se debe considerar los ángulos de proyección del dispositivo y la distancia máxima a la cual se debe ubicar el mismo.



**Figura 15-3:** Ejemplos resueltos de ubicación del Mando Wii

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

De la misma forma se puede realizar cálculos similares con el eje horizontal para obtener distancias de separación o calcular ángulos de desplazamiento en el mismo eje, por lo tanto se puede calcular el alcance que podrá cubrir el sistema de acuerdo a los requerimientos establecidos o en el lugar donde se quiera implementar.

### 3.5.2 Pruebas Realizadas

Para calibrar la pantalla se toma alrededor de 5 minutos. Como procedimiento colocamos el Mando Wii en la base en dirección a la pantalla proyectada por el infocus, en este punto elegimos el ángulo de azimut y de elevación. Como podemos apreciar en la Figura 16-3 se describe de forma gráfica lo antes mencionado.

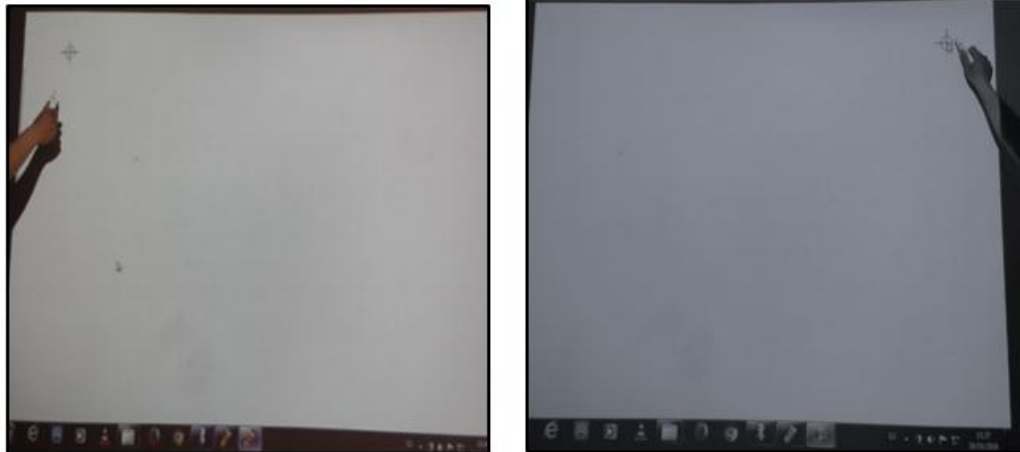
Una vez ubicado el dispositivo se debe evitar moverlo, esto garantizara su funcionamiento y obtención de mejores resultados.



**Figura 16-3:** Fijación del Mando Wii en la base

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

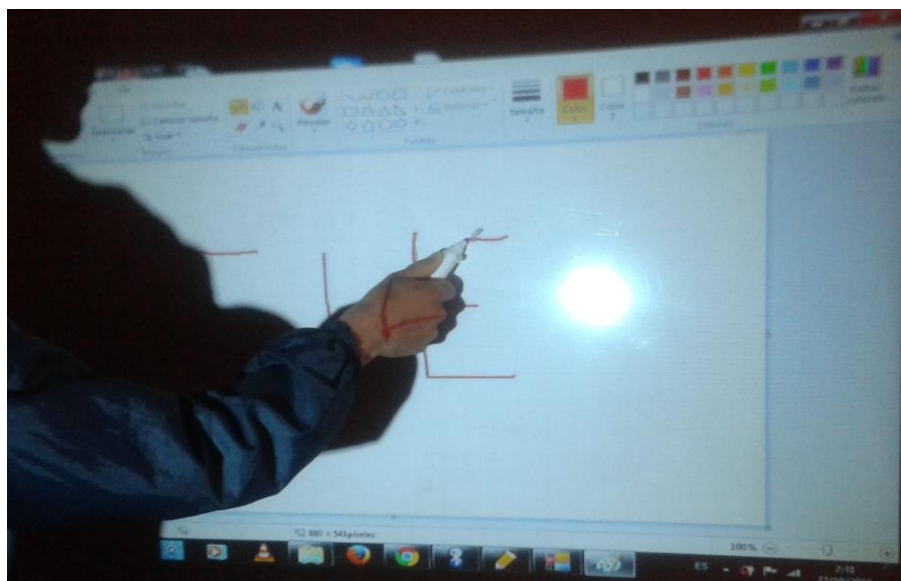
Con el puntero infrarrojo tocamos la figura color rojo tipo cruz ubicada en la parte superior izquierda presionando el pulsador incorporado el cual enciende el diodo IR., este procedimiento se repite en los demás puntos ubicados en la parte superior derecha, inferior izquierda y derecha.



**Figura 17-3:** Calibración de la pantalla

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

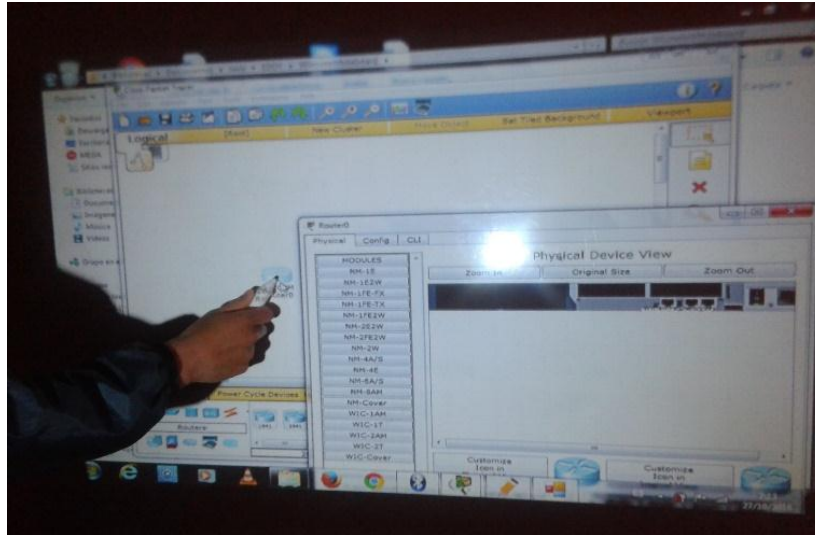
Una vez realizado los pasos anteriores, podemos empezar a utilizar la pantalla digital interactiva, como ejemplo abrimos Paint y dibujamos las iniciales (FIE) Figura 18-3, comprobando su eficiencia y eficacia ya que el retardo/latencia no se puede percibir, es decir en este caso se dibuja en tiempo real y mediante la percepción se puede considerar que tiempo requerido para ejecutar esta instrucción es similar al realizarlo con el cursor.



**Figura 18-3:** Función de arrastre

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

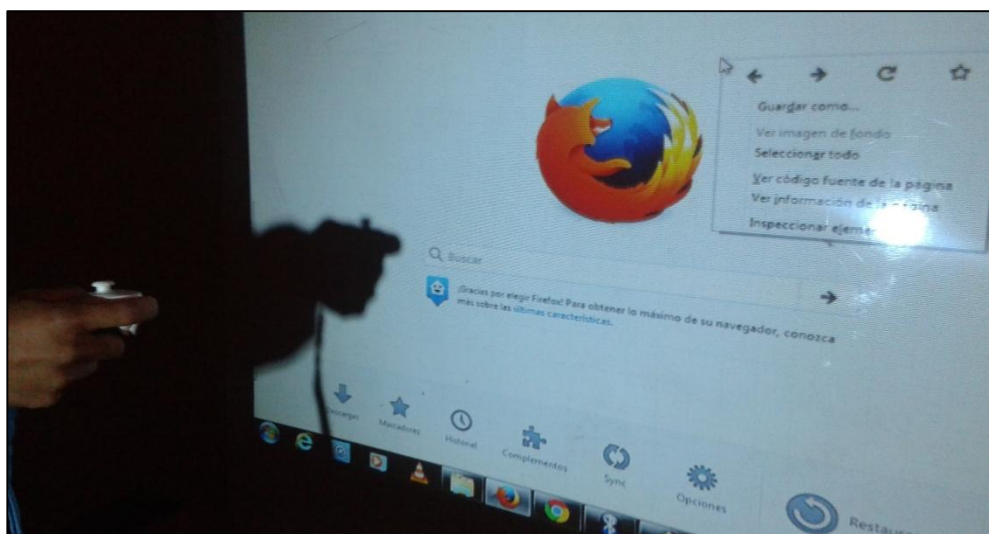
De la misma forma comprobamos la opción de clic izquierdo, el cual se ejecuta al presionar el botón del puntero, al pulsar dos veces ejecuta la función de doble clic, al igual que la función de arrastrar mostrada en la imagen anterior, la precisión y el tiempo requerido para la realización de estas instrucciones es muy eficiente.



**Figura 19-3:** Comprobación del puntero

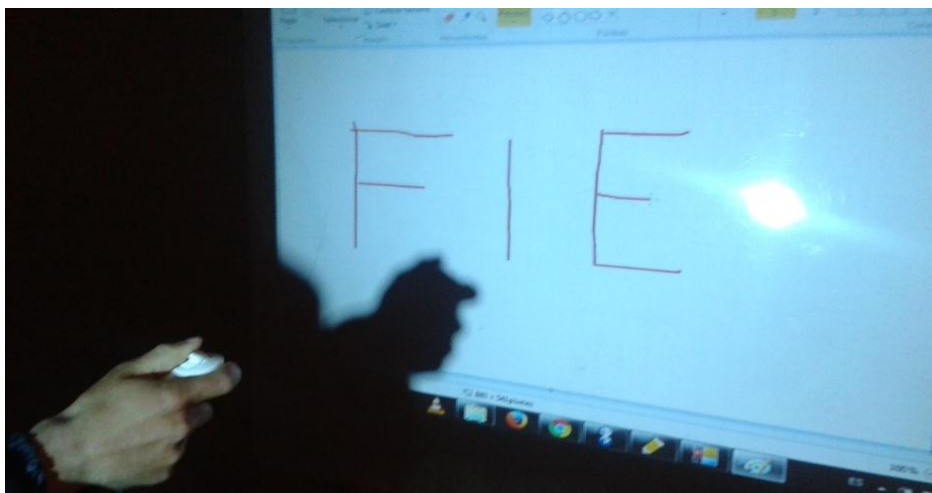
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

Para comprobar el funcionamiento del nunchuk (Figura 20-3) damos clic en la segunda opción del menú “INGRESAR” luego que la aplicación reconoce el dispositivo, como ejemplo ejecutamos el programa Cisco Packet Tracer y probamos todas las opciones del programa como lo hacemos con el mouse tradicional, y efectivamente funciona como tal, la latencia es mínima lo que hace de este un dispositivo excelente para su uso.



**Figura 20-3:** Función clic derecho del control.

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

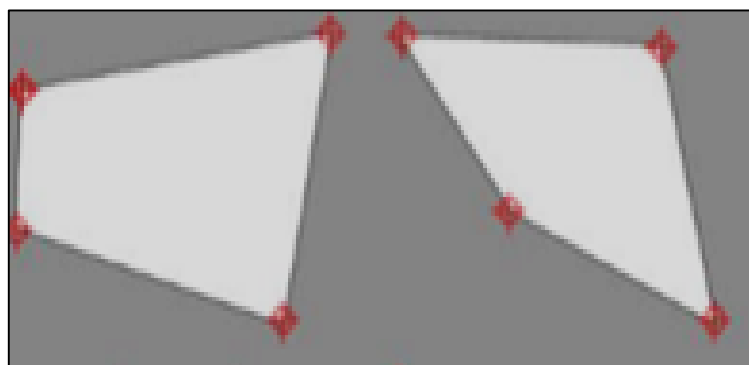


**Figura 21-3:** Función clic izquierdo del control

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

### 3.6 Análisis de Resultados

Para utilizar la pizarra interactiva se debe calibrar presionando los 4 puntos que se generan al iniciar la aplicación, los errores más comunes es el ángulo y la distancia en la cual es ubicado el Mando Wii, movimientos accidentales al dispositivo y la inadecuada proyección de la pantalla sobre la superficie en la cual vamos a trabajar.



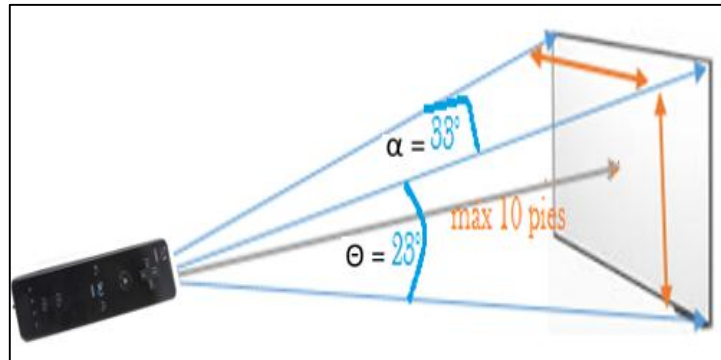
**Figura 22-3:** Ejemplo de pantallas mal proyectadas

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Por lo cual la proyección de la pantalla debe ser lo más centrada posible y equilibrada en sus extremos con la facilidad de variar la dimensión de la misma, evitando inclinaciones y deformaciones con lo cual se puede reducir considerablemente la posibilidad de instrucciones fallidas, y mediante un soporte mantener fijo el dispositivo infrarrojo en la posición en la cual se desee utilizar.



Además, en esta etapa consideramos el ángulo de visión de la cámara infrarroja tanto de forma vertical ( $\theta$ ) como horizontal ( $\alpha$ ) y la distancia máxima a la que se debe ubicar el dispositivo en relación a la superficie proyectada, dentro de estos rangos el Mando Wii puede captar el punto de luz emitido por el puntero infrarrojo, si excede estos parámetros el sistema no funcionará correctamente.



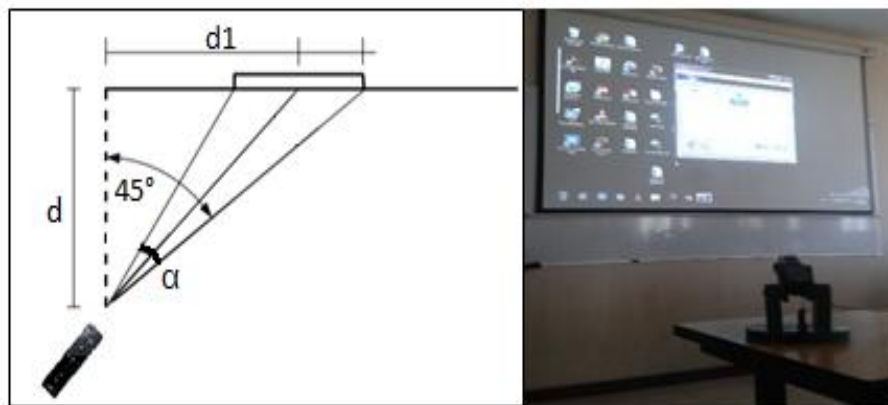
**Figura 23-3:** Ángulos y distancia de funcionamiento del Mando Wii.

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Con el fin de obtener información y datos que demuestren cuales son las condiciones y requerimientos con las cuales la detección de los puntos mediante la cámara infrarroja del dispositivo sea las mejores se realizaron mediciones y ensayos.

Para realizar las pruebas correspondientes se realizó 40 peticiones con el puntero IR en diferentes ángulos y radios respecto de la superficie proyectada, para realizar el cálculo dividimos el valor asignado para 120 y sacamos el valor porcentual equivalente.

- Mando Wii a 45 grados de la pizarra



**Figura 24-3:** Enfoque a 45°

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

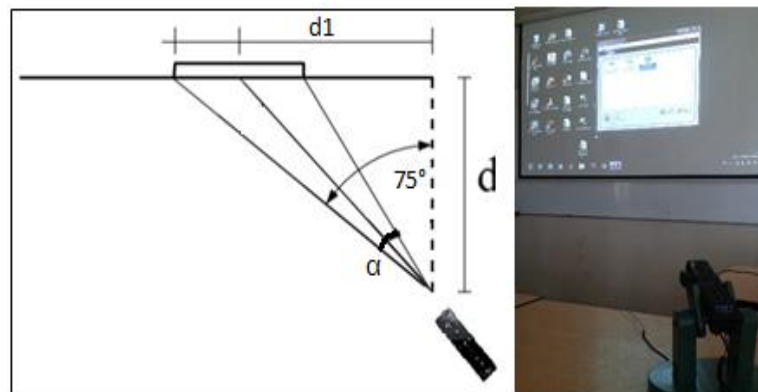
**Tabla 1-3:** Datos a 45 grados.

Distancia	Bueno		Aceptable		Malo		Total	
2m	8,3%	10	24,2%	29	0,8%	1	33,3%	40
3m	<u>30,0%</u>	36	2,5%	3	0,8%	1	33,3%	40
4m	1,7%	2	4,2%	5	27,5%	33	33,3%	40
<b>Equivalente</b>	40,0%		30,8%		29,2%		100,0%	120

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Tomando en cuenta el valor más alto como un funcionamiento adecuado, en este ángulo los resultados son muy buenos ya que los obstáculos se reducen mejorando la línea de vista entre la cámara IR y el puntero, la distancia recomendable es la de 3m.

- Mando Wii a 75 grados de la pizarra



**Figura 25-3:** Enfoque a 75°

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

**Tabla 2-3:** Datos a 75 grados.

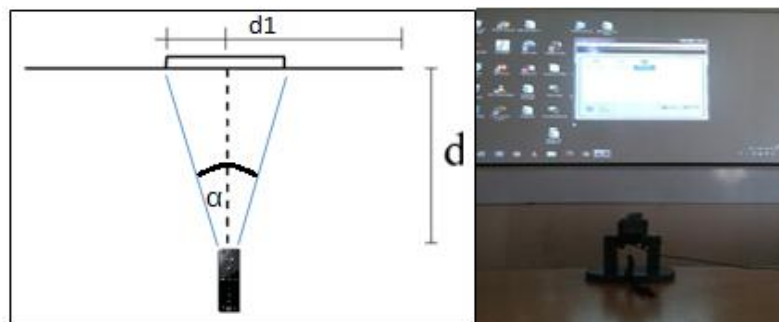
Distancia	Bueno		Aceptable		Malo		Total	
2m	2,5%	3	20,8%	25	10,0%	12	33,3%	40
3m	4,2%	5	<u>24,2%</u>	29	5,0%	6	33,3%	40
4m	0,0%	0	2,5%	3	30,8%	37	33,3%	40
<b>Equivalente</b>	6,7%		47,5%		45,8%		100,0%	120

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016

Como siguiente opción tenemos a 75°, esta ubicación tiene más desventajas que las anteriores ya que el cuerpo del usuario obstaculiza la línea de vista entre los dispositivos interrumpiendo la comunicación, la distancia recomendable es la de 3m.



- Mando Wii a 90 grados de la pizarra.



**Figura 26-3:** Enfoque a 90°

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016

**Tabla 3-3:** Datos a 90 grados.

Distancia	Bueno		Aceptable		Malo		Total	
2m	6,7%	8	14,2%	17	12,5%	15	33,3%	40
3m	10,0%	12	<u>18,3%</u>	22	5,0%	6	33,3%	40
4m	0,8%	1	10,0%	12	22,5%	27	33,3%	40
<b>Equivalente</b>	17,5%		42,5%		40,0%		100,0%	120

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Como segunda opción tenemos la ubicación en 90°, en este caso el valor porcentual influye más en aceptable debido que la probabilidad de obstáculos aumenta un poco más que en 45°, la distancia recomendable es la de 3m.

### 3.7 Validación de Resultados

Para su respectiva validación se consideró los siguientes parámetros:

#### 3.7.1 Por sus características

Se realizó la comparación con diferentes PDIs existentes, donde se consideró las características y funcionalidades más importantes que deben poseer estas pantallas para lograr un entorno interactivo y dinámico, entre estos se consideró lo siguiente:

En la tabla 4-3 Se pondero con un uno (1) cuando la PDI en cuestión presenta dicha característica y caso contrario con un (0), dichas características se basaron en la PDI Smart Board 885i6 considerada con las mejores prestaciones actualmente.

**Tabla 4-3:** Características de las PDI en el mercado

<b>Características</b>	<b>PDI Propuesta</b>	<b>PDI Mimio Tech</b>	<b>PDI Qomo</b>	<b>PDI Smart Board 885i6</b>
Software de la PDI	1	1	1	1
Funciones de mouse	1	1	1	1
Funciones táctiles	0	1	1	1
Rotuladores o punteros	1	1	1	1
Control inalámbrico	1	0	0	1
Aplicaciones adicionales	0	0	1	1
Accesorios adicionales	1	1	1	1
<b>Total</b>	<b>5</b>	<b>5</b>	<b>6</b>	<b>7</b>
<b>Porcentaje</b>	<b>71.4%</b>	<b>71.4%</b>	<b>85.7%</b>	<b>100 %</b>

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016

La PDI que posee las mejores características es la Smart Board 885i6 como se mencionó anteriormente obteniendo un porcentaje del 100 %, seguido de la marca Qomo quien obtuvo un 85.7% y la Mimio Tech con un 71,4% equivaliendo a unas PDI con buenas prestaciones, por otro lado tenemos la PDI propuesta quien a pesar de no tener Funciones táctiles y aplicaciones adicionales, cuenta con las características más importantes de las existentes en el mercado obteniendo un valor aceptable de 71.4%.

### **3.7.2 Por su precio**

Uno de las razones principales para el desarrollo del presente proyecto se basó en construir un sistema que disponga de características y funciones similares a una Pantalla Digital Interactiva comercial, el costo de este elemento es considerable por lo que es poco accesible económicamente para las personas o entidades que están interesadas en su adquisición.

Para el diseño y desarrollo de este sistema se consideró principalmente este aspecto, desde la selección del dispositivo principal hasta el último componente necesario del proyecto, por lo cual

en esta etapa se evaluara el costo total que se necesitara para su elaboración, los cuales se muestran a continuación:

**Tabla 5-3:** Materiales y costo de la Pizarra Interactiva

Material	Costo Aproximado
Puntero Infrarrojo	\$ 5,00
Wii Mando	\$ 40,00
Bluetooth	\$ 10,00
Soporte	\$ 10,00
Pilas	\$ 2,00
<b>Total:</b>	<b>\$ 67,00</b>

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

**Tabla 6-3:** Materiales y costo del Control Inalámbrico

Material	Costo Aproximado
Nunchuk	\$ 20,00
Arduino Nano	\$ 13,00
2 Módulos Xbee	\$ 90,00
Carcasa	\$ 2,00
<b>Total:</b>	<b>\$ 125,00</b>

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

En la estimación del costo total necesario para el proceso de construcción de la Pizarra Digital Interactiva con control inalámbrico propuesto no se considera elementos como proyector, pizarra y computador debido a que son dispositivos complementarios que están disponibles en una entidad educativa; por lo tanto su costo sería aproximadamente:

**Tabla 7-3:** Costo total de la PDI propuesta

Elementos	Costo
Pizarra Interactiva	\$ 67,00
Control Inalámbrico	\$ 125,00
Otros	\$ 5,00
<b>Total:</b>	<b>\$ 197,00</b>

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

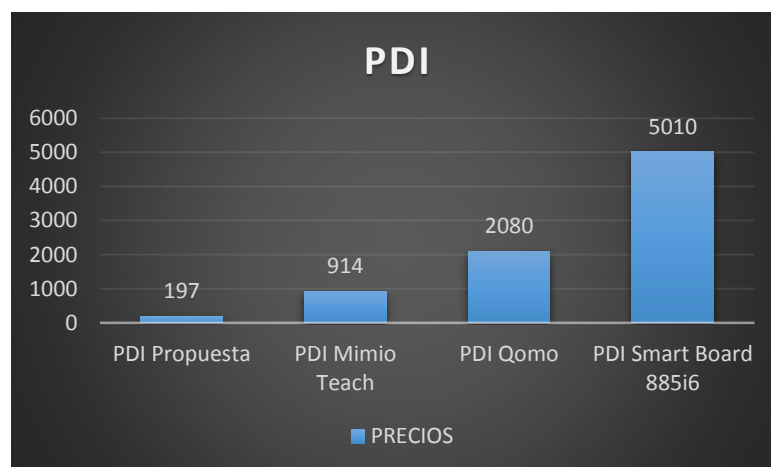
Debido al uso de materiales de bajo costo y de fácil adquisición los mismos que se mantienen vigentes con actualizaciones y soportes constantes en caso de tecnologías, nuestra pizarra interactiva tendría bastante competitividad con respecto a una comercial, contando con la mayoría de las funcionalidades y características de una existente.

Se realizó una búsqueda en el mercado libre de nuestro país y se encontró algunos modelos de pizarras interactivas cuyo precio varía de acuerdo a la marca y funcionalidades que presentan, además de los componentes adicionales que se incluyen por su compra, se eligió 3 modelos y a continuación se muestra un cuadro comparativo entre ellos:

**Tabla 8-3: Precios de PDI en el mercado**

PDI	Precio
PDI Mimio Teach	\$ 914,00
PDI Qomo	\$ 2080,00
PDI Smart Board 885i6	\$ 5010,00

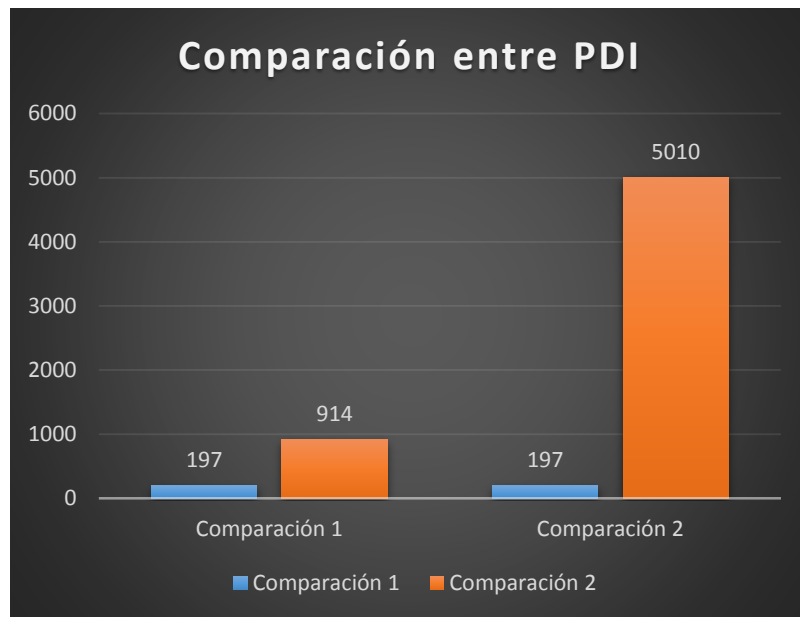
**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.



**Gráfico 1-3: Cuadro comparativo de precios**

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

Para la validación del proyecto, en el Gráfico 3-1, claramente se puede observar una gran diferencia en relación al costo, al compararse con los otros tres alternativas es sumamente más económico independientemente de los componentes que lo conformen, ya que el funcionamiento y los beneficios que presentan son similares tanto con el uso de una PDI comercial con la PDI propuesta.



**Gráfico 2-3:** Comparación entre dos PDI

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

En la comparación 1, se visualiza la diferencia de costo que existe entre nuestra PDI propuesta y la PDI comercial con el precio más bajo entre las cuatro alternativas, con un cálculo simple se puede decir que la adquisición de la Pizarra Digital Interactiva tan sería un 21,55% de su valor.

En la comparación 2, esta relación aumenta ya que es la PDI comercial con el precio más elevado, por lo cual existe una diferencia más considerable, la implementación del sistema propuesto ahorraría un 96,07% del valor de la PDI más cara.

### 3.7.3 *Por su eficiencia y precisión.*

Para la validación de este punto se realizó una encuesta (ver ANEXO D) a los alumnos del instituto IPREX, para lo cual se tomó una muestra de 80 personas, la misma que consta de 6 preguntas basados en diferentes campos de acuerdo al objetivo que se requiere evaluar.

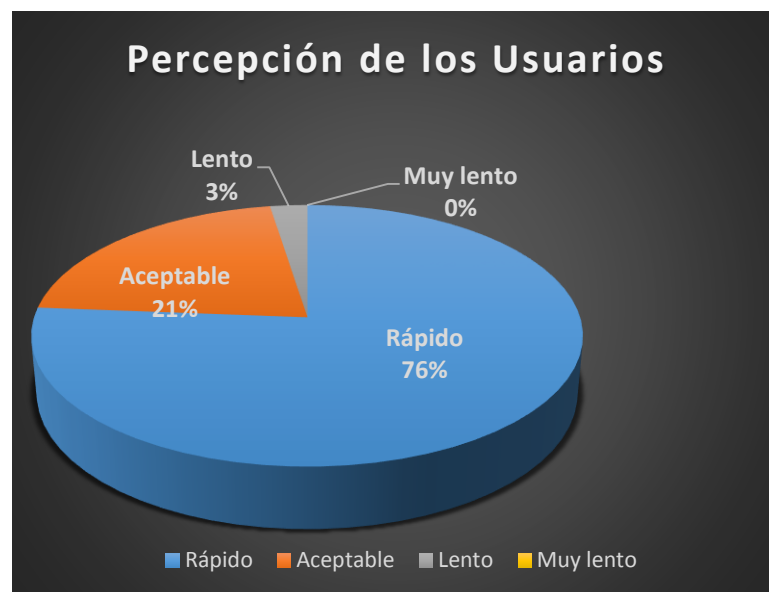
Una vez implementado la PDI propuesta y explicado su funcionamiento, mediante la manipulación del sistema, observaciones y actividades desarrolladas con los alumnos se obtuvieron los siguientes resultados en la encuesta:

- En la primera pregunta sobre el tiempo de respuesta que se tarda la PDI propuesta en ejecutar la instrucción del puntero infrarrojo, se obtuvieron los siguientes valores:

**Tabla 9-3:** Tiempo de respuesta de la PDI

Percepción	Nº Alumnos	Porcentaje
Rápido	61	76.3%
Aceptable	17	21.3%
Lento	2	2.5%
Muy lento	0	0.0%
<b>Total</b>	<b>80</b>	<b>100.0%</b>

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



**Gráfico 3-3:** Percepción de los Usuarios

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Como se aprecia en la Tabla 9-3 a un bajo porcentaje del 2,5% de las personas encuestadas al usar este prototipo les pareció lento, sin embargo la percepción que obtuvieron la mayoría de los estudiantes con respecto al tiempo de respuesta fue rápida (76,3%), seguido de un pequeño número de alumnos que les pareció una respuesta aceptable (21,3%) dando una sensación similar a la ejecución de una instrucción realizada directamente del ratón.

- En la segunda pregunta se verifico su correcto funcionamiento, consultando si las acciones realizadas con el puntero infrarrojo corresponden a las instrucciones requeridas por el usuario, se obtuvieron los siguientes resultados:

**Tabla 10-3:** Funcionamiento de la PDI

Ejecución	Nº Alumnos	Porcentaje
Si	68	85.0%
No	12	15.0%
<b>Total</b>	<b>80</b>	<b>100.0%</b>

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

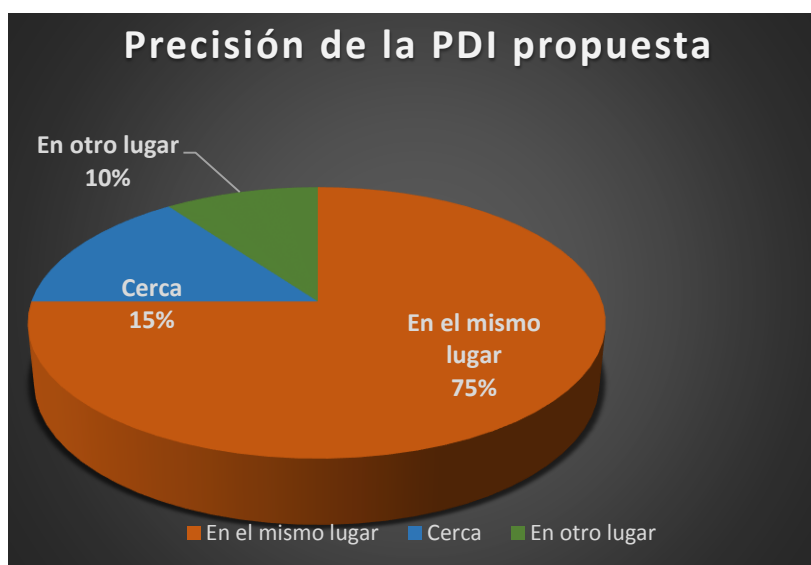
La mayoría de los alumnos validaron con una respuesta positiva, dando a conocer que la aplicación que interpreta y ejecuta las instrucciones entre el mando wii y el ordenador ha sido desarrollada correctamente.

- En la tercera pregunta se buscó comprobar la precisión al utilizar el puntero IR, para lo cual se consultó si la instrucción realizada por el mismo se dio en el lugar señalado, ante esta interrogante se obtuvo lo siguiente:

**Tabla 11-3:** Precisión de la PDI

Ubicación	Nº Alumnos	Porcentaje
En el mismo lugar	60	75.0%
Cerca	12	15.0%
En otro lugar	8	10.0%
<b>Total</b>	<b>80</b>	<b>100.0%</b>

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



**Gráfico 4-3:** Ubicación del cursor

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Como resultado un 75% de la muestra indico que las acciones realizadas con el puntero infrarrojo se realizaron correctamente dando a entender que se ejecutó en el mismo lugar señalado, obteniendo una precisión considerablemente alta, por otro lado es importante realizar correctamente la calibración del sistema para así evitar imprecisiones al momento de usar la PDI.

En este caso este procedimiento fue realizado tanto por los autores y los estudiantes, obteniendo un 10% de errores.

- En la cuarta pregunta se verifico el funcionamiento del control inalámbrico.

**Tabla 12-3:** Funcionamiento del Nunchuk

Ejecución con C.I	Nº Alumnos	Porcentaje
Si	68	85.0%
No	12	15.0%
<b>Total</b>	80	100.0%

**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

De todos los participantes, 85% de alumnos corroboraron el buen funcionamiento del nunchuk y su sistema inalámbrico dando como resultado un buen desempeño del dispositivo; un 15 % de personas estuvieron en desacuerdo debido a problemas de distancia y conexiones que no consideraron.

Para cuantificar o medir el grado de eficiencia y presión de nuestra PDI propuesta hemos considerado los resultados de estas cuatro preguntas, debido a que la muestra es un número constante y para el análisis de estos datos se consideraron criterios cuantitativos, hemos seleccionado la media aritmética de los valores más relevantes dentro de estas preguntas, dando como resultado un 80,3% de eficiencia que garantiza el buen funcionamiento de nuestro proyecto.

#### **3.7.4 Por su nivel de aceptación**

En este punto tratamos sobre el impacto generado en el usuario, y los resultados como recurso didáctico en la enseñanza y aprendizaje, para esto se realizaron preguntas a los alumnos dentro de la encuesta anterior.

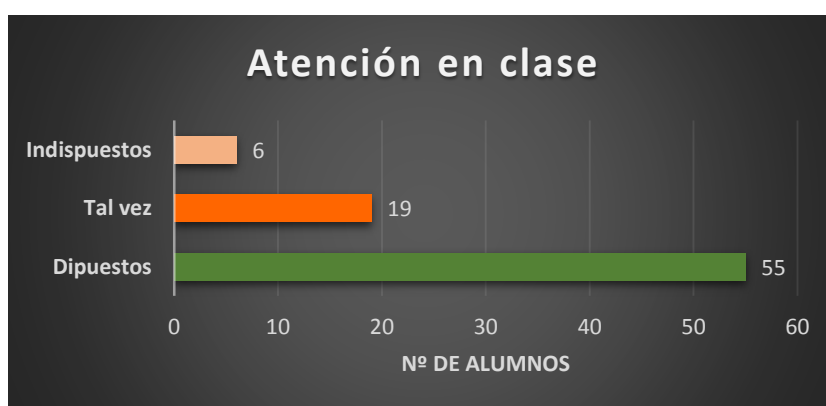


- En la quinta pregunta, se buscó saber si el alumno estaría dispuesto a prestar más atención a la clase, lo cual se comprobó en la siguiente tabla:

**Tabla 13-3:** Atención a la clase

Decisión	Nº Alumnos	Porcentaje
Si	55	68,8%
Tal vez	19	23,8%
No	6	7,5%
<b>Total</b>	<b>80</b>	<b>100,0%</b>

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



**Gráfico 5-3:** Resultados del impacto en la clase.

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

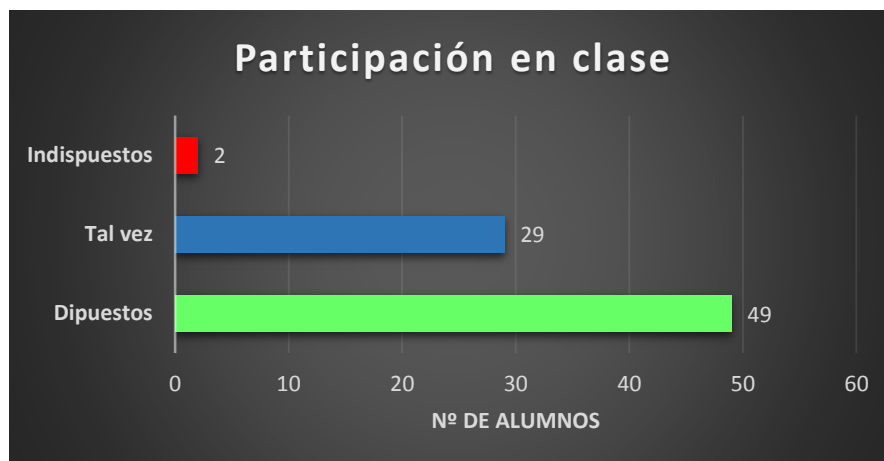
Debido al no estar familiarizado con la PDI propuesta un pequeño porcentaje se mostró indiferente al uso de este sistema, sin embargo entre los que se mantuvieron indecisos y a los alumnos que mostraron mayor interés estaban de acuerdo en poner más atención a las clases impartidas por ser el entorno más dinámico.

- En la sexta pregunta, se trató de comprobar el aumento de participación en clase, se obtuvo los siguientes resultados:

**Tabla 14-3:** Participación en clase

Decisión	Nº Alumnos	Porcentaje
Si	49	61,3%
Tal vez	29	36,3%
No	2	2,5%
<b>Total</b>	<b>80</b>	<b>100,0%</b>

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.



**Gráfico 6-3:** Resultados del impacto en la clase.

Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

Antes de la obtención de resultados, el docente encargado manifestó que pocos alumnos voluntariamente participan en clase, sin embargo luego de la implantación y manipulación de la PDI propuesta el número de estudiantes dispuestos a participar en clase incremento hasta alcanzar la mayoría de ellos.

Con los resultados anteriores podemos verificar que la implementación de nuestro sistema tiene un alto grado aceptación y produce un impacto positivo en el proceso de enseñanza y aprendizaje de los usuarios debido a que genera un ambiente mucho más dinámico e interactivo incentivando a los alumnos a una mayor participación y una mejor concentración en clases obteniendo mejores resultados en el proceso.

## CONCLUSIONES

- Al realizar varias pruebas en diferentes ángulos, se concluye que la mejor desviación de la señal es a 45° ya que es el más óptimo para obtener mayor precisión y confiabilidad.
- Realizado el análisis costo beneficio, la pizarra digital interactiva emulada es un 80% más económica que las pizarras comerciales existentes en el mercado, obteniendo un resultado casi similar a dichas pantallas, logrando así el acceso a estos dispositivos por su bajo precio.
- La pantalla interactiva realizado en este trabajo de titulación cuenta con un componente adicional que otras pantallas interactivas de bajo costo no poseen, se trata de un control inalámbrico que tiene clic derecho e izquierdo simulando un mouse inalámbrico.
- Luego de haber utilizado diferentes lenguajes de programación se determina que Labview es mejor para este tipo de proyectos ya que se logró diseñar y programar una aplicación capaz de interpretar la información proveniente del Mando Wii y del Nunchuk inalámbrico.
- Realizando un breve análisis de las características de los diferentes módulos Xbee decidimos utilizar el modelo S1 ya que cuenta con una cobertura de 100m suficiente para realizar las pruebas necesarias de nuestro prototipo, a diferencia de otros modelos que cuentan con coberturas en el rango de Kilómetros sería innecesario su uso en el presente trabajo.

## RECOMENDACIONES

- Construir un soporte giratorio, el cual nos permita colocar el Mando Wii y enfocar a la superficie plana siguiendo los pasos antes mencionados.
- Para calibrar la pantalla se debe realizarlo cuidadosamente, para ello nos guiamos en las figuras en forma de cruz tocando con el puntero infrarrojo los más al centro posible de la cruz, de esta forma logramos mayor exactitud y precisión.
- Se recomienda que el mando o la pantalla proyectada en la superficie sufran algún tipo de movimiento, debe volver a calibrar la pantalla, visualmente se aprecia una descoordinación entre los dispositivos.
- Se debe utilizar una superficie reflectiva para aumentar el brillo del puntero infrarrojo y así tener más precisión.
- Revisar la opción Test/Query del software X-CTU para verificar si el ordenador detectó el dispositivo Xbee porque en algunos casos no se encuentran instalados los drivers adecuados, firmware del ordenador, etc.

## BIBLIOGRAFÍA

1. **ARDUINO.CC.** *Arduino Nano*. [En línea], [Consulta: 28 enero 2016], Disponible en: [tps://www.arduino.cc/en/Main/ArduinoBoardNano](https://www.arduino.cc/en/Main/ArduinoBoardNano)
2. **ARDUINO.CC.** *Introduction*. [En línea], [Consulta: 20 febrero 2016], Disponible en: <https://www.arduino.cc/en/Guide/Introduction>
3. **ASENCIO, G et al.** “Interoperabilidad en sistemas domóticos mediante pasarela infrarrojos-ZigBee”. *Revista iberoamericana de automática e informática industrial* [en línea]. 2011. [Consulta: 20 octubre 2015]. Disponible en: [http://apps.elsevier.es/watermark/ctl\\_servlet?\\_f=10&pident\\_articulo=90029254&pident\\_usuario=0&pcontactid=&pident\\_revista=331&ty=148&accion=L&origen=zonadelectura&web=www.elsevier.es&lan=es&fichero=331v08n04a90029254pdf001.pdf](http://apps.elsevier.es/watermark/ctl_servlet?_f=10&pident_articulo=90029254&pident_usuario=0&pcontactid=&pident_revista=331&ty=148&accion=L&origen=zonadelectura&web=www.elsevier.es&lan=es&fichero=331v08n04a90029254pdf001.pdf)
4. **CARRIÓN, J, & TORRES. R.** *Implementación de una pizarra digital interactiva multiusuario con desarrollo de aplicación para Tele Educación en personas con capacidades diferentes*. (En línea) (Tesis pregrado). Escuela Politécnica del Ejercito, Departamento de Eléctrica y Electrónica, Sangolquí, Ecuador, 2012. p 43, Disponible en: <http://repositorio.espe.edu.ec/bitstream/21000/5442/1/T-ESPE-033352.pdf>
5. **DE LOPE, J.** *Emisión y Recepción de Infrarrojos*. [En línea]. 2001. [Consulta: 15 Julio 2015]. Disponible en: <http://www.dia.fi.upm.es/~jdllope/docs/delope01a.pdf>
6. **FERNÁNDEZ, R.** “La pizarra digital interactiva como una de las tecnologías emergentes en la enseñanza”, *<sup>3</sup>Ciencia*, [en línea], 2012, Cuba, pp 5-7, [consulta: 10 Marzo 2016], ISSN: 2227-2690, Disponible en: <http://revistas.unica.cu/index.php/uciencia/article/view/1/1>
7. **FERNÁNDEZ, R.** *La pizarra digital interactiva como una de las tecnologías emergentes en la enseñanza actual*, Universidad & Ciencia, [en línea], 2013, Cuba, pp 6-8, [4 de Marzo de 2016], Disponible en: <https://dialnet.unirioja.es/descarga/articulo/4817356.pdf>
8. **GONSALEZ, J.** “El lenguaje de programación C#”, *Introducción a C#*, 2014, pp 7-8 Colombia, [En línea], [Consulta: 2016-09-28], Disponible en: <http://dis.um.es/~bmoros/privado/bibliografia/LibroCsharp.pdf>

9. **GONZÁLEZ. C, RODRIGO PRADINES PINO,** Análisis de software para desarrollo entorno gráfico Labview y propuesta de implementación para laboratorio en el instituto de electricidad y electrónica en universidad austral de chile. (En línea) (Tesis Pregrado), Universidad Austral de Chile, Valdivia, 2007, [Consulta: 2016-08-10], Disponible en:  
<http://cybertesis.uach.cl/tesis/uach/2007/bmfcig643a/sources/bmfcig643a.pdf>
  
10. **LA WIIMOTELIB,** [en línea] [Consulta: 20-09-2016], disponible en:  
<http://bibing.us.es/proyectos/abreproy/11823/fichero/Volumen+I%252F4.pdf>
  
11. **MEDINA. E.** *Sistema de control de inventarios y facturación de productos utilizando etiquetas de radiofrecuencia (RFID) para tecnilibro.* (En línea) (Tesis pregrado), Escuela Politécnica del Ejercito, Facultad de Ingeniería en Sistemas e Informática, pp 25-27, [Consulta: 2016-08-20], Disponible en:  
<http://repositorio.espe.edu.ec/bitstream/21000/631/1/T-ESPE-012138.pdf>
  
12. **MERCADO LIBRE,** *Pizarra Digital Interactiva Qomo 88 Pulgadas,* [En Línea], [Consulta: 5 Septiembre 2016], Disponible en: [http://articulo.mercadolibre.com.ec/MEC-410151753-pizarra-digital-interactiva-qomo-88-pulgadas-\\_JM](http://articulo.mercadolibre.com.ec/MEC-410151753-pizarra-digital-interactiva-qomo-88-pulgadas-_JM)
  
13. **MERCADO LIBRE,** *Pizarra Digital Interactiva Smart Board 885i6 87',* [En Línea], [Consulta: 4 Septiembre 2016], Disponible en: [http://articulo.mercadolibre.com.ec/MEC-410081253-pizarra-digital-interactiva-smart-board-885i6-87-\\_JM](http://articulo.mercadolibre.com.ec/MEC-410081253-pizarra-digital-interactiva-smart-board-885i6-87-_JM)
  
14. **MERCADO LIBRE,** *Pizarra Interactiva Portatil Mimio Teach* [En Línea], [Consulta: 5 Septiembre 2016], Disponible en: [http://articulo.mercadolibre.com.ec/MEC-410003920-pizarra-interactiva-portatil-mimio-teach-\\_JM](http://articulo.mercadolibre.com.ec/MEC-410003920-pizarra-interactiva-portatil-mimio-teach-_JM)
  
15. **NATIONAL INSTRUMENTS.** *Call Library Function Node,* [en línea], [Consulta: 25 enero 2016], Disponible en: [http://zone.ni.com/reference/en-XX/help/371361H-01/glang/call\\_library\\_function/](http://zone.ni.com/reference/en-XX/help/371361H-01/glang/call_library_function/)
  
16. **NATIONAL INSTRUMENTS.** *Estructuras de Ejecución en LabVIEW,* [en línea], [Consulta: 20 enero 2016], Disponible en: <http://www.ni.com/getting-started/labview-basics/esa/execution-structures>

17. **NATIONAL INSTRUMENTS.** *Instrumentación virtual con Labview*, [en línea], [Consulta: 20 enero 2016], Disponible en: <http://www.etitudela.com/entrenadorcomunicaciones/downloads/labviewcursodeinstrumentacion.pdf>
  
18. **NATIONAL INSTRUMENTS,** *LabVIEW 2015*, [en línea], [Consulta: 1 diciembre 2016], Disponible en: <http://www.ni.com/newsroom/release/write-code-faster-with-labview-2015/es/>
  
19. **PARUELO, J.** “La caracterización funcional de ecosistemas mediante sensores remotos”. *Revista Ecosistemas*, [en línea], 2008, [consulta: 20 de agosto 2016], Disponible en: <http://revistaecosistemas.net/index.php/ecosistemas/article/view/83>
  
20. **SAMANIEGO. R,** *Wiimote Hack*, [En línea], (Tesis Pregrado), Escuela Técnica Superior Ingeniería 2009. [Consulta: 20 diciembre 2015], Disponible en: <http://deeea.urv.cat/public/PROPOSTES/pub/pdf/1231pub.pdf>
  
21. **Tutorial de Labview,** [En línea], [Consulta: 2016-10-28], Disponible en: <http://www.esi2.us.es/~asun/LCPC06/TutorialLabview.pdf>
  
22. **VALERO. R,** *La integración del mando inalámbrico Wii Remote dentro de la arquitectura software YARP*, [en línea] (Tesis Pregrado), Universidad Carlos III de Madrid, Leganés, España, 2010, pp 21-24, [Consulta: 26 Marzo 2016] Disponible en: [http://www.repositoriocrdnet.net:8080/bitstream/handle/123456789/651/Tes\\_RodrigoValeroR\\_Integraci%C3%B3nMandoInalambrico\\_2010.pdf?sequence=1](http://www.repositoriocrdnet.net:8080/bitstream/handle/123456789/651/Tes_RodrigoValeroR_Integraci%C3%B3nMandoInalambrico_2010.pdf?sequence=1)

## ANEXOS

### ANEXO A: Cálculo de ángulo de proyección

Para la solución del ejemplo tratado en el análisis de resultados, utilizamos los parámetros de visualización de la cámara infrarroja del Wii Mando en este caso utilizaremos el ángulo de proyección vertical que es de  $26^\circ$ .

$$\tan 13 = \frac{x}{3m}$$

$$x = (\tan 13) \cdot 3m$$

$$x = 0,69 \, m$$

$$\tan x = \frac{0,4 \, m}{3 \, m}$$

$$x = \tan^{-1}\left(\frac{0,4}{3}\right)$$

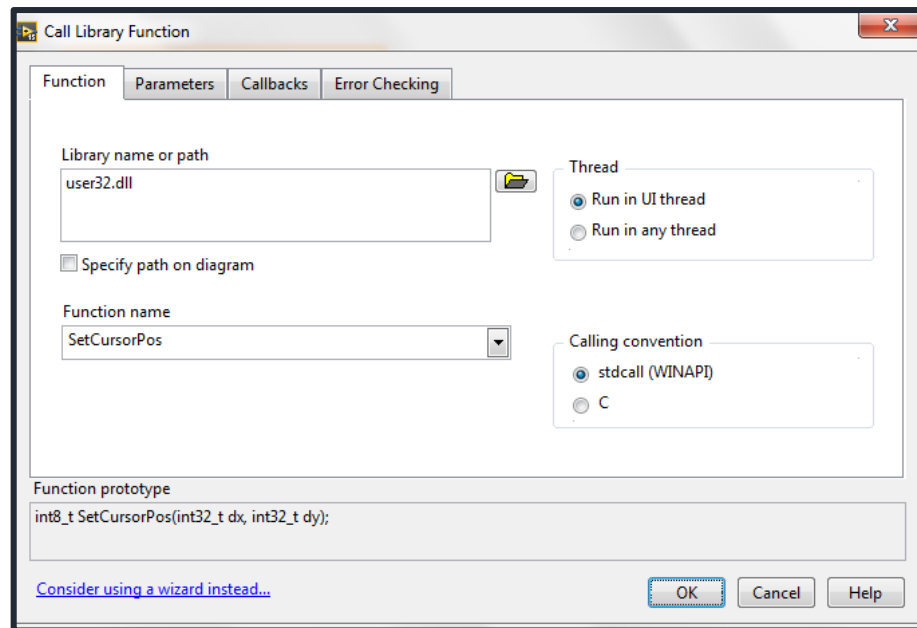
$$x = 0,4m$$



## ANEXO B: Configuración de la función Call Library Function Node.

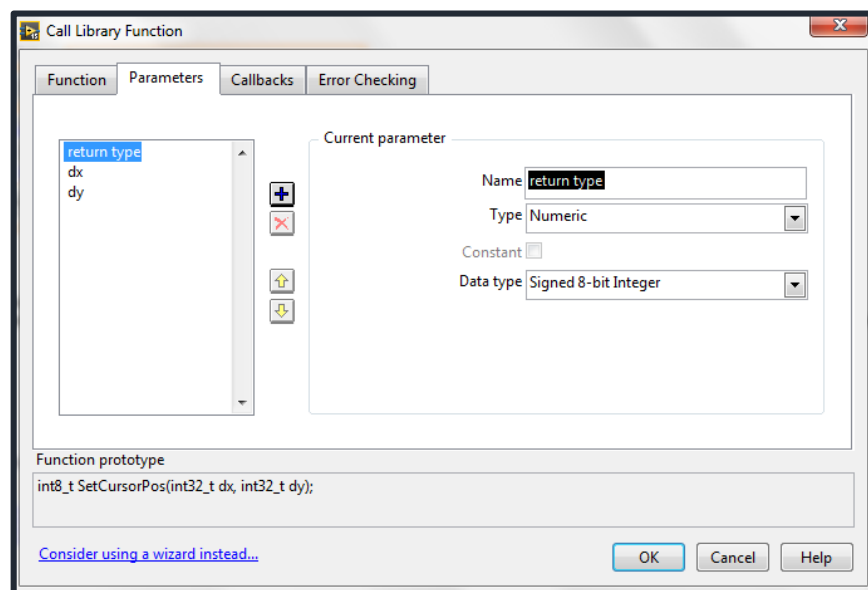
Utilizamos la librería user32.dll del sistema operativo.

- ✓ Selección de la función del movimiento del cursor.



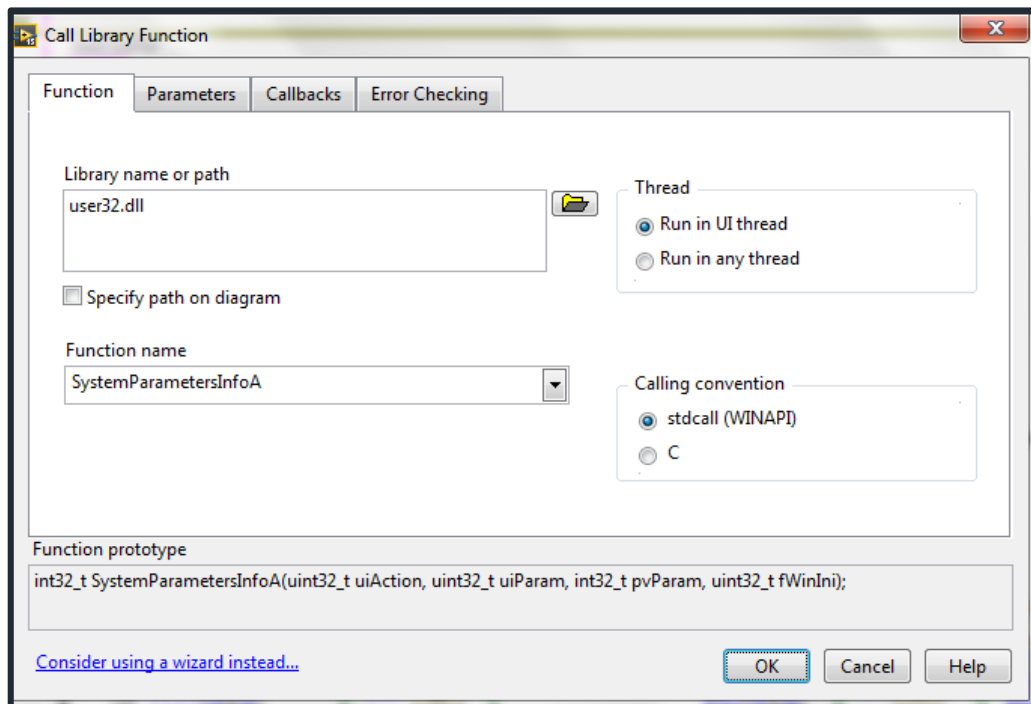
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

- ✓ Configuración de los parámetros de Call Library Function Node.



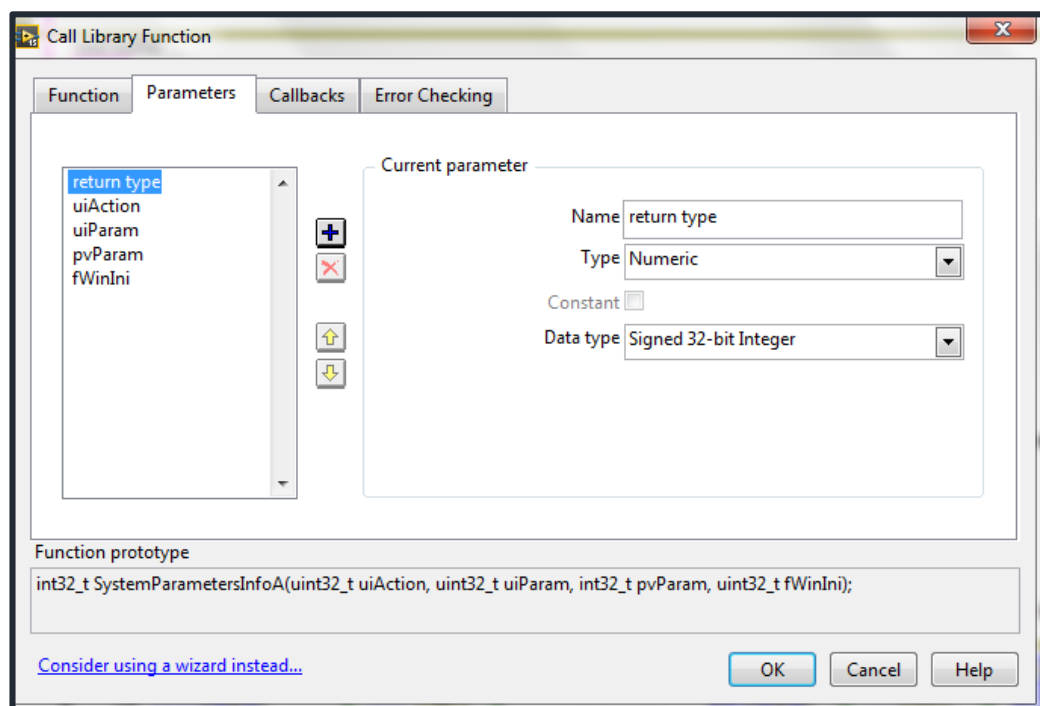
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

- ✓ Para los eventos de clic.



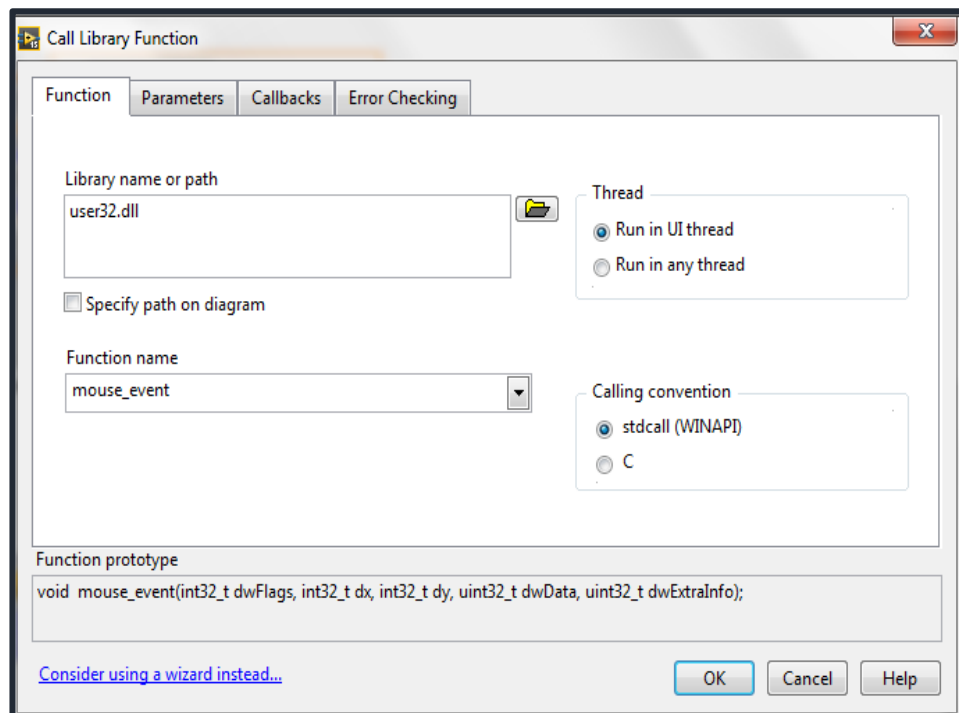
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

- ✓ Configuración de parámetros para enlazar eventos de clic.



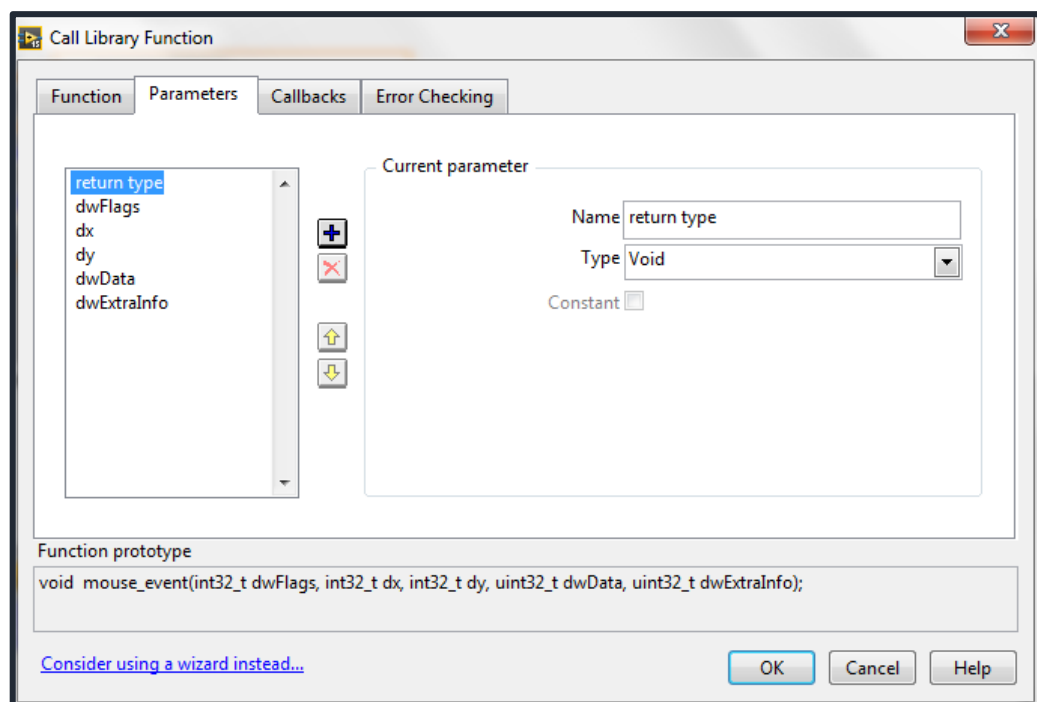
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

- ✓ Selección de la función de evento de ratón.



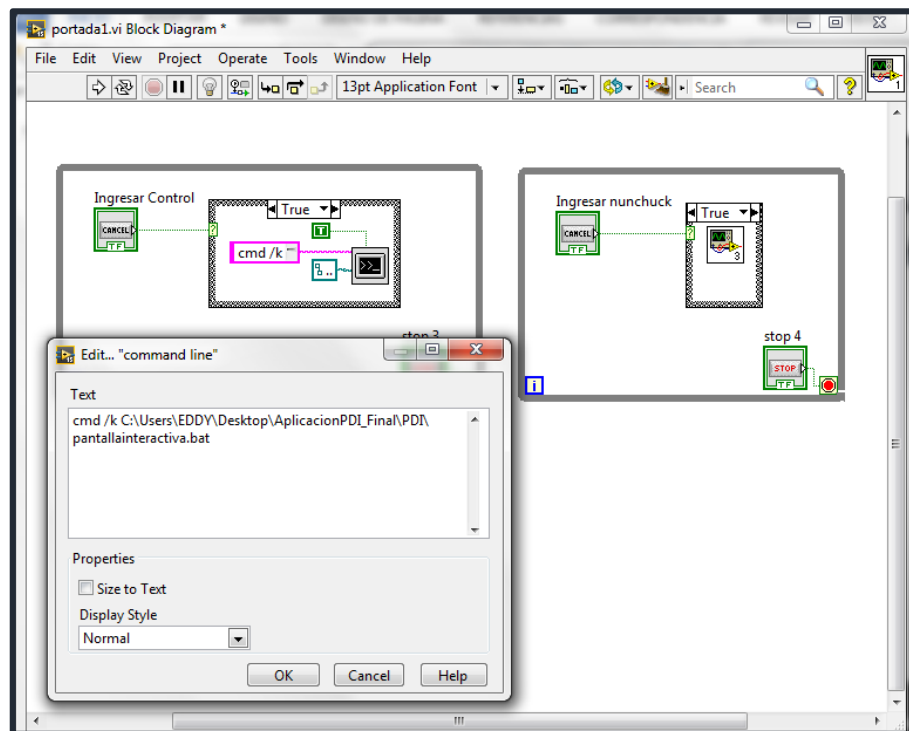
Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

- ✓ Configuración de parámetros para evento de ratón.



Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

- ✓ Configuración de la función System Exec.vi



Realizado por: Jorge Zúñiga; Eddy Gusqui, 2016.

## ANEXO C: Alimentación directa del Mando Wii.

El Mando Wii Plus requiere baterías para su funcionamiento sin embargo el consumo que requiere el dispositivo es muy excesivo agotando rápidamente las mismas, para realizar pruebas con el mando se creó un circuito regulador de voltaje, el cual nos permite obtener el mismo resultado que ofrece la batería pero con la diferencia que se conecta directamente a un tomacorriente de 110v.



**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

Los elementos y dispositivos utilizados para la implementación del circuito regulador de tensión son: una baquelita, un capacitor cerámico 0.1uf (103), un capacitor electrolítico de 1uf, una resistencia de 240 ohm, un LM317 y un cargador de celular que actúa como transformador.



**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

De igual forma para el puntero infrarrojo, construimos un circuito similar al del Mando Wii hasta realizar las pruebas correspondientes.



**Realizado por:** Jorge Zúñiga; Eddy Gusqui, 2016.

## ESCUELA SUPERIOR POLITECNICA DE CHIMBORAZO

### FACULTAD DE INFORMATICA Y ELECTRÓNICA

#### ESSCUELA DE ELECTRÓNICA EN TELECOMUNICACIONES Y REDES

### ENCUESTA ACERCA DE LA EFECTIVIDAD DE LA PDI PROPUESTA

1.- para usted el tiempo de respuesta de la PDI propuesta al realizar una instrucción (clic, doble clic, arrastre, etc.) es:

RAPIDO.....      ACEPTABLE.....      LENTO.....      MUY LENTO.....

2.- ¿Al ejecutar una acción con el puntero infrarrojo o el control inalámbrico se ejecuta la instrucción requerida?

SI\_\_\_\_\_

NO\_\_\_\_\_

3.- Al acercar el puntero infrarrojo en cualquier punto sobre la pantalla proyectada, ¿Dónde se ejecuta la instrucción del puntero?

En el mismo lugar\_\_\_\_\_

Cerca\_\_\_\_\_

En otro lugar\_\_\_\_\_

4.- Al utilizar el control inalámbrico ¿Se ejecutan las instrucciones deseadas?

SI\_\_\_\_\_

NO\_\_\_\_\_

5.- ¿Prestarías más atención a la clase si el entorno es más dinámico e interactivo?

SI\_\_\_\_\_

NO\_\_\_\_\_

6.- ¿Participarías más en clase al utilizar la PDI propuesta?

SI\_\_\_\_\_

NO\_\_\_\_\_